

# Strong Oracle Optimality of Folded Concave Penalized Estimation

Jianqing Fan, Lingzhou Xue and Hui Zou

Princeton University and University of Minnesota

This Version: October 20th, 2012

## Abstract

Folded concave penalization methods (Fan and Li, 2001) have been shown to enjoy the strong oracle property for high-dimensional sparse estimation. However, a folded concave penalization problem usually has multiple local solutions and the oracle property is established only for one of the unknown local solutions. A challenging fundamental issue still remains that it is not clear whether the local optimal solution computed by a given optimization algorithm possesses those nice theoretical properties. To close this important theoretical gap in over a decade, we provide a unified theory to show explicitly how to obtain the oracle solution using the local linear approximation algorithm. For a folded concave penalized estimation problem, we show that as long as the problem is localizable and the oracle estimator is well behaved, we can obtain the oracle estimator by using the one-step local linear approximation. In addition, once the oracle estimator is obtained, the local linear approximation algorithm converges, namely produces the same estimator in the next iteration. The general theory is demonstrated by using three classical sparse estimation problems, i.e. the sparse linear regression, the sparse logistic regression and the sparse precision matrix estimation, where the LASSO penalized least squares, the LASSO penalized logistic regression and the CLIME are used as the initial estimator, respectively.

**Key Words:** Folded concave penalty; Local linear approximation; Non-convex optimization; Oracle estimator; Sparse estimation; Strong oracle property.

# 1 Introduction

Sparse estimation is at the center of the stage of high-dimensional statistical learning. The two mainstream methods are the LASSO (or  $\ell_1$  penalization) and the folded concave penalization (Fan and Li, 2001) such as the SCAD and the MCP. Numerous papers have been devoted to the numerical and theoretical study of both methods. A strong irrepresentable condition is necessary for the LASSO to be selection consistent (Zhao and Yu, 2006; Zou, 2006; Meinshausen and Bühlmann, 2006). The folded concave penalization, unlike the LASSO, does not require the irrepresentable condition to achieve selection consistency and can correct the intrinsic estimation bias of the LASSO penalization (Fan and Li, 2001; Fan and Peng, 2004; Zhang, 2010a; Fan and Lv, 2011). The LASSO owns its popularity largely to its computational properties. For certain learning problems, such as the LASSO penalized least squares, the solution paths are piecewise linear which allows one to employ a LARS-type algorithm to compute the entire solution path efficiently (Efron et al., 2004). For a more general class of LASSO penalized problems, the coordinate descent algorithm has been shown to be very useful and efficient (Friedman et al., 2008, 2010).

The computation for folded concave penalized methods is much more involved, because the resulting optimization problem is usually non-convex and has multiple local minimizers. Several algorithms have been developed for computing the folded concave penalized estimators. Fan and Li (2001) worked out the local quadratic approximation (LQA) algorithm as a unified method for computing the folded concave penalized maximum likelihood. Zou and Li (2008) proposed the local linear approximation (LLA) algorithm which turns a concave penalized problem into a series of reweighted  $\ell_1$  penalization problems. Both LQA and LLA are related to the MM principle (Hunter and Lange, 2004; Hunter and Li, 2005). Zhang (2010a) devised a PLUS algorithm for solving the penalized least squares using the MCP and the SCAD. Recently, coordinate descent was applied to solve the folded concave penalized least squares (Mazumder et al., 2011; Fan and Lv, 2011). With these advances in computing algorithms, one can now at least efficiently compute a local solution of the folded concave penalized problem. It has been shown repeatedly that the folded concave penalty performs better than the LASSO in various high-dimensional sparse estimation problems. Examples include sparse linear regression model estimation (Fan and Li, 2001; Zhang, 2010a), sparse

generalized linear model estimation (Fan and Lv, 2011), sparse Cox’s proportional hazards model estimation (Bradic et al., 2012), sparse precision matrix estimation (Lam and Fan, 2009) and sparse Ising model estimation (Xue et al., 2012), among others.

Before declaring that the folded concave penalization is superior to the LASSO, we still need to resolve a missing puzzle in the picture. The optimal theoretical properties of the folded concave penalization are established for a theoretic local solution. However, we have to employ one of these local minimization algorithms to find such a local solution. It still remains to prove that the computed local solution is the desired theoretic local solution to make the theory fully relevant. Many have tried to address this issue (Zhang, 2010a; Fan and Lv, 2011; Zhang and Zhang, 2012). The basic idea there is to find conditions under which the folded concave penalized problem actually has a unique minimizer and hence eliminate the problem of multiple local solutions. Although this line of thoughts is very natural and logically intuitive, the imposed conditions for the unique minimizer are too strong to be realistic.

In this paper we offer a very different and direct approach to deal with the multiple local solutions issue. We outline a general procedure based on the LLA algorithm for computing a specific local solution of the folded concave penalization problem and then derive a lower bound on the probability that this specific computed solution exactly equals to the oracle solution. This probability lower bound equals  $1 - \delta_0 - \delta_1 - \delta_2$  where  $\delta_0$  corresponds to the localizability of the underlying model,  $\delta_1$  and  $\delta_2$  represent the regularity of the oracle estimator and they have nothing to do with any actual estimation method. Explicit expressions of  $\delta_0$ ,  $\delta_1$  and  $\delta_2$  are given in Section 2. Under weak regularity conditions,  $\delta_1$  and  $\delta_2$  are very small. Thus, if  $\delta_0$  goes to zero then the computed LLA solution is the oracle estimator with an overwhelming probability. On the other hand, if  $\delta_0$  cannot go to zero then it means that the underlying model is extremely difficult to estimate no matter how clever an estimator is. Therefore, our theory suggests a “bet-on-folded-concave-penalization” principle, since as long as there is a reasonable estimator our procedure can deliver an optimal estimator using the folded concave penalization via the one-step LLA implementation. Furthermore, we use concrete examples to show how to prove all tail probabilities  $\delta_0$ ,  $\delta_1$  and  $\delta_2$  go to zero at a fast rate under the ultra-high dimensional setting where  $\log(p) = O(n^\eta)$  for some  $0 < \eta < 1$ .

Throughout this paper the following useful notation will be used. For a matrix  $\mathbf{U} = (u_{ij})$ ,

denote by  $\|\mathbf{U}\|_{\min} = \min_{(i,j)} |u_{ij}|$  the minimum absolute value, and denote by  $\lambda_{\min}(\mathbf{U})$  and  $\lambda_{\max}(\mathbf{U})$  the smallest and largest eigenvalues of  $\mathbf{U}$ , respectively. We also use several matrix norms: the  $\ell_1$  norm  $\|\mathbf{U}\|_{\ell_1} = \max_j \sum_i |u_{ij}|$ , the  $\ell_2$  norm  $\|\mathbf{U}\|_{\ell_2} = \sqrt{\lambda_{\max}(\mathbf{U}'\mathbf{U})}$ , the  $\ell_\infty$  norm  $\|\mathbf{U}\|_{\ell_\infty} = \max_i \sum_j |u_{ij}|$ , the entry-wise  $\ell_1$  norm  $\|\mathbf{U}\|_1 = \sum_{(i,j)} |u_{ij}|$  and the entry-wise  $\ell_\infty$  norm  $\|\mathbf{U}\|_{\max} = \max_{(i,j)} |u_{ij}|$ . For any symmetric matrix, its  $\ell_1$  norm is equal to its  $\ell_\infty$  norm.

## 2 Main Results

We begin with a somewhat abstract/general presentation of the sparse estimation problem. Consider estimating a model based on  $n$  independent and identically distributed  $p$ -dimensional observations, where the feature dimension  $p$  is much larger than the sample size  $n$ . The target of estimation is a  $p$ -dimensional “parameter”  $\boldsymbol{\beta}^* = (\beta_1^*, \dots, \beta_p^*)'$ , that is, the underlying model is parameterized by  $\boldsymbol{\beta}^*$ . Remark that in some problems the target of estimation  $\boldsymbol{\beta}^*$  can be a matrix (e.g., a covariance matrix). In such cases it is understood that  $(\beta_1^*, \dots, \beta_p^*)'$  is the vectorization of the matrix  $\boldsymbol{\beta}^*$ . Denote its corresponding support set as  $\mathcal{A} = \{j : \beta_j^* \neq 0\}$  with the cardinality to be  $s = |\mathcal{A}|$ . The sparsity assumption means that  $s \ll p$ .

Suppose that our estimation scheme is to get a local minimizer of the penalized convex loss function problem

$$\min_{\boldsymbol{\beta}} \ell_n(\boldsymbol{\beta}) + P_\lambda(|\boldsymbol{\beta}|), \quad (1)$$

where  $\ell_n(\boldsymbol{\beta})$  represents the convex loss function and  $P_\lambda(|\boldsymbol{\beta}|) = \sum_j P_\lambda(|\beta_j|)$  is a folded concave penalty function. The above formulation is a bit abstract but covers many important statistical models and estimators. For example,  $\ell_n(\boldsymbol{\beta})$  can be the squared error loss in penalized least squares and the negative log-quasi-likelihood function in penalized maximum quasi-likelihood.

An oracle knows the true support set  $\mathcal{A}$  of the underlying model and the oracle estimator is defined as

$$\widehat{\boldsymbol{\beta}}^{oracle} = (\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle}, \mathbf{0}) = \arg \min_{\boldsymbol{\beta}_{\mathcal{A}}, \boldsymbol{\beta}_{\mathcal{A}^c} = \mathbf{0}} \ell_n(\boldsymbol{\beta}). \quad (2)$$

We assume throughout the paper that the problem is regular so that the oracle solution is

unique, satisfying

$$\nabla_j \ell_n(\hat{\boldsymbol{\beta}}^{oracle}) = 0, \quad \forall j \in \mathcal{A} \quad (3)$$

where  $\nabla_j$  is the partial derivative with the  $j^{th}$  component of  $\boldsymbol{\beta}$ . Note that the oracle estimator is not a feasible estimator but it can be used as a theoretical benchmark for other estimators to compare with. An estimator is said to have the oracle property if the estimator and the oracle estimator have the same asymptotic distribution (Fan and Li, 2001; Fan and Peng, 2004). Moreover, an estimator is said to have the strong oracle property if the estimator equals the oracle estimator with an overwhelming probability (Kim, et. al, 2008; Fan and Lv, 2011).

Throughout this paper, we only need to assume that  $\ell_n(\cdot)$  is a differentiable convex function, and we also assume that  $P_\lambda(|t|) = P_{a,\lambda}(|t|)$  is a general folded concave penalty function defined on  $t \in (-\infty, \infty)$  satisfying

- (i)  $P_\lambda(t)$  is increasing and concave in  $t \in [0, \infty)$ ;
- (ii)  $P_\lambda(t)$  is differentiable in  $t \in (0, \infty)$  with  $P'_\lambda(0) := P'_\lambda(0+) \geq a_1 \lambda$ ;
- (iii)  $P'_\lambda(t) \geq a_1 \lambda$  for  $t \in (0, a_2 \lambda]$ ;
- (iv)  $P'_\lambda(t) = 0$  for  $t \in [a \lambda, \infty)$  with the pre-specified constant  $a > a_2$ .

where  $a_1$  and  $a_2$  are some fixed positive constants. Note the definition follows and extends previous works on SCAD and MCP (Fan and Li, 2001; Zhang, 2010a; Fan and Lv, 2011). Folded concave penalty was introduced to bridge the  $\ell_1$  penalty and the  $\ell_0$  penalty (Fan and Li, 2001). On the interval  $[-a_2 \lambda, a_2 \lambda]$ , the desired penalty should penalize small coefficients as the  $\ell_1$  penalty, and on the intervals outside the interval  $(-a \lambda, a \lambda)$ , the penalty function should behave more like the  $\ell_0$  penalty to avoid introducing biases. The above family of general folded concave penalties has included several popular concave penalties proposed in recent years, for example the SCAD (Fan and Li, 2001) whose derivative is given by

$$P'_\lambda(t) = \lambda I_{\{t \leq \lambda\}} + \frac{(a \lambda - t)_+}{a - 1} I_{\{t > \lambda\}}, \quad \text{for some } a > 2,$$

and the MCP (Zhang, 2010a) whose derivative is given by

$$P'_\lambda(t) = (\lambda - \frac{t}{a})_+, \quad \text{for some } a > 1.$$

By simple calculation, it is easy to see that  $a_1 = a_2 = 1$  for the SCAD and  $a_1 = 1 - a^{-1}$ ,  $a_2 = 1$  for the MCP.

Numerical results have been provided in the statistical literature to show that the folded concave penalty performs much better than the  $\ell_1$  penalty in terms of both model estimation accuracy and variable selection consistency. To offer theoretical understanding of their differences, it is important to show that the obtained local solution of the fold concave penalized estimator has better theoretical properties than the LASSO estimator. However, a general technical difficulty in the folded concave regularization problems is to show that the computed local solution is the local solution with proven theoretical properties. Under strong conditions, it has been argued that the folded concave penalized problem has a unique minimizer and hence any algorithm finding a local solution will find the global minimizer (Zhang, 2010a; Fan and Lv, 2011; Zhang and Zhang, 2012). The problem with this argument is that in reality it is very rare that the folded concave penalized problem actually has a unique minimizer, which in turn implies that these strong conditions are too stringent to hold in practice. See the numerical results in Section 4.

We argue that, although the estimator is defined via a folded concave penalization problem, we only care about the properties of computed estimator. It is perfectly fine that the computed local solution is not the global minimizer, as long as it has the optimal or desired statistical properties. In this paper we directly analyze a specific solution by the local linear approximation (LLA) algorithm (Zou and Li, 2008). The LLA algorithm takes advantage of the special folded concave structure of penalty functions and utilizes the majorization-minimization principle to turn a concave regularization problem into a sequence of weighted  $\ell_1$  penalization problems. Within each iteration of the LLA algorithm, the underlying local linear approximation is actually the best convex majorization of the concave penalty function (see Theorem 2 of Zou and Li (2008)). Moreover, the majorization-minimization principle has provided theoretical justification to guarantee the convergence of the LLA algorithm to a stationary point of the concave regularization problem (1). The LLA convex relaxation idea has been used in Candes et al. (2008), Zhang (2010b), Fan and Lv (2011), Bradic et al. (2012) and Huang and Zhang (2012).

Here, we summarize the details of the LLA algorithm as in Algorithm 1.

**Remark 1.** Zhang (2010b) gave a high-dimensional analysis of the LLA algorithm in the

---

**Algorithm 1** The LLA algorithm

---

1. Initialize  $\hat{\boldsymbol{\beta}}^{(0)} = \hat{\boldsymbol{\beta}}^{initial}$  and compute the adaptive weight

$$\hat{\mathbf{w}}^{(0)} = (\hat{w}_1^{(0)}, \dots, \hat{w}_p^{(0)})' = \left( P'_\lambda(|\hat{\beta}_1^{(0)}|), \dots, P'_\lambda(|\hat{\beta}_p^{(0)}|) \right)'.$$

2. For  $m = 1, 2, \dots$ , repeat the LLA iteration till convergence

- (2.a) Obtain  $\hat{\boldsymbol{\beta}}^{(m)}$  by solving the following optimization problem

$$\hat{\boldsymbol{\beta}}^{(m)} = \min_{\boldsymbol{\beta}} \ell_n(\boldsymbol{\beta}) + \sum_j \hat{w}_j^{(m-1)} \cdot |\beta_j|,$$

- (2.b) Update the adaptive weight vector  $\hat{\mathbf{w}}^{(m)}$  with  $\hat{w}_j^{(m)} = P'_\lambda(|\hat{\beta}_j^{(m)}|)$ .
- 

linear regression models, and Huang and Zhang (2012) further provided a detailed technical analysis of the LLA algorithm in the high-dimensional generalized linear models. Huang and Zhang (2012) required the convex loss function  $\ell_n(\boldsymbol{\beta})$  to be twice differentiable, and the theoretical results critically depend on the complex general invertibility factor, c.f. Definition 3 of Huang and Zhang (2012). In this work, we consider a more general folded concave penalized convex loss problem without requiring  $\ell_n(\boldsymbol{\beta})$  to be twice differentiable, and we discuss how the LLA algorithm can actually find the oracle estimator (2) with an overwhelming probability under fairly weak regularity conditions. Especially, our high-dimensional analysis does not depend on the complex general invertibility factor as in Huang and Zhang (2012).

In the following theorems, we provide the non-asymptotic analysis of the LLA algorithm for obtaining the oracle estimator  $\hat{\boldsymbol{\beta}}^{oracle}$  in the folded concave penalized problem if it is initiated by some initial estimator  $\hat{\boldsymbol{\beta}}^{initial}$ . To simplify notation, we define  $\nabla \ell_n(\boldsymbol{\beta}) = (\nabla_1 \ell_n(\boldsymbol{\beta}), \dots, \nabla_p \ell_n(\boldsymbol{\beta}))$  as the gradient vector of  $\ell_n(\boldsymbol{\beta})$ . Moreover, denote by  $\mathcal{A}^c$  the complement of the true support set  $\mathcal{A}$ , i.e.  $\mathcal{A}^c = \{j : \beta_j^* = 0\}$ , and set  $\nabla_{\mathcal{A}^c} \ell_n(\boldsymbol{\beta}) = (\nabla_j \ell_n(\boldsymbol{\beta}) : j \in \mathcal{A}^c)$  with respect to  $\mathcal{A}^c$ .

**Theorem 1.** *Suppose the minimal signal strength of  $\boldsymbol{\beta}^*$  satisfies that*

$$(A\theta) \quad \|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} > (a+1)\lambda.$$

Consider the folded concave penalized problem with  $P_\lambda(\cdot)$  satisfying (i)–(iv). Let  $a_0 = \min\{1, a_2\}$ . Under the event

$$\mathcal{E}_1 = \left\{ \|\hat{\beta}^{initial} - \beta^*\|_{\max} \leq a_0 \lambda \right\} \cap \left\{ \|\nabla_{\mathcal{A}^c} \ell_n(\hat{\beta}^{oracle})\|_{\max} < a_1 \lambda \right\},$$

the LLA algorithm initiated by  $\hat{\beta}^{initial}$  finds the oracle estimator  $\hat{\beta}^{oracle}$  after one iteration.

Applying the union bound to  $\mathcal{E}_1$ , we easily get the following corollary.

**Corollary 1.** *With a probability at least  $1 - \delta_0 - \delta_1$ , the LLA algorithm initiated by  $\hat{\beta}^{initial}$  finds the oracle estimator  $\hat{\beta}^{oracle}$  after one iteration, where*

$$\delta_0 = \Pr \left( \|\hat{\beta}^{initial} - \beta^*\|_{\max} > a_0 \lambda \right)$$

and

$$\delta_1 = \Pr \left( \|\nabla_{\mathcal{A}^c} \ell_n(\hat{\beta}^{oracle})\|_{\max} \geq a_1 \lambda \right).$$

**Remark 2.** By its definition,  $\delta_0$  represents the localizability of the underlying model. To apply Theorem 1 we need to have an appropriate initial estimator to make  $\delta_0$  go to zero as  $n$  and  $p$  diverge to infinity, namely the underlying problem is localizable. In Section 3 we will show by concrete examples that how to find a good initial estimator to make the problem localizable.  $\delta_1$  represents the regularity behavior of the oracle estimator, i.e., its closeness to the true “parameter” measured by the score function. Note that  $\nabla_{\mathcal{A}^c} \ell_n(\beta^*)$  is concentrated around zero. Thus,  $\delta_1$  is usually small.

In summary, Theorem 1 and its corollary state that as long as the problem is localizable and regular, we can find an oracle estimator by using one-step local linear approximation, which can be regarded as the generalization of the LLA algorithm and the one-step estimation idea (Zou and Li, 2008) to the high-dimensional setting.

**Theorem 2.** *Consider the folded concave penalized problem with  $P_\lambda(\cdot)$  satisfying (i)–(iv). Under the event*

$$\mathcal{E}_2 = \left\{ \|\nabla_{\mathcal{A}^c} \ell_n(\hat{\beta}^{oracle})\|_{\max} < a_1 \lambda \right\} \cap \left\{ \|\hat{\beta}_{\mathcal{A}}^{oracle}\|_{\min} > a \lambda \right\},$$

as long as the LLA algorithm finds the oracle estimator  $\hat{\beta}^{oracle}$ , the LLA algorithm will find  $\hat{\beta}^{oracle}$  again in the next iteration, i.e. the LLA algorithm converges to  $\hat{\beta}^{oracle}$  in the next iteration.



Now we combine Theorems 1 and 2 to derive the non-asymptotic probability bound for the LLA algorithm to exactly converge to the oracle estimator  $\hat{\beta}^{oracle}$  in the general folded concave penalized problem (1).

**Corollary 2.** *Consider the folded concave penalized problem with  $P_\lambda(\cdot)$  satisfying (i)–(iv). Under the assumption of (A0), the LLA algorithm initiated by  $\hat{\beta}^{initial}$  converges to the oracle estimator  $\hat{\beta}^{oracle}$  after two iterations with a probability at least  $1 - \delta_0 - \delta_1 - \delta_2$ , where*

$$\delta_2 = \Pr \left( \|\hat{\beta}_{\mathcal{A}}^{oracle}\|_{\min} \leq a\lambda \right).$$

**Remark 3.** The localizable probability  $1 - \delta_0$  and regularity probability  $1 - \delta_1$  have been defined before.  $\delta_2$  is a probability on the magnitude of the oracle estimator. Both  $\delta_1$  and  $\delta_2$  are related to the regularity behavior of the oracle estimator and will be referred to the oracle regularity condition. Under the minimum signal condition (A0), it requires only the uniform convergence of  $\hat{\beta}_{\mathcal{A}}^{oracle}$ . Namely,

$$\delta_2 \leq \Pr \left( \|\hat{\beta}_{\mathcal{A}}^{oracle} - \beta_{\mathcal{A}}^*\|_{\max} > \lambda \right).$$

Thus we can regard  $\delta_2$  as a direct measurement of the closeness of the oracle estimator to the true “parameter” and is usually small because of a small intrinsic dimensionality  $s$ . This will indeed be shown in Section 3.

### 3 Theoretical Examples

In the sequel, we outline three classical examples to demonstrate interesting and powerful applications of Theorems 1 and 2 to solve folded concave penalization problems. We need basically to check the localizable condition and the regularity condition for these problems.

We focus specifically on the least-squares, logistic regression, and sparse precision matrix estimation to derive a more explicit bound and to give cleaner results and proofs. For more general cases in the family of the generalized linear models, the localizable condition  $\delta_0$  using LASSO can be verified by using the result of Fan and Lv (2011) and the regularity conditions can be verified by using the concentration inequality of the maximum likelihood estimator in Fan and Song (2010).

### 3.1 Sparse linear regression

The first example is the canonical problem of the folded concave penalized least square estimation, i.e.

$$\min_{\boldsymbol{\beta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_{\ell_2}^2 + \sum_j P_\lambda(|\beta_j|) \quad (4)$$

where  $\mathbf{y} \in \mathbb{R}^n$  and  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)' \in \mathbb{R}^{n \times p}$ . Let  $\boldsymbol{\beta}^*$  be the true parameter vector in the linear regression model  $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \varepsilon$ , and then the true support set of  $\boldsymbol{\beta}^* = (\beta_j^*)_{1 \leq j \leq p}$  is  $\mathcal{A} = \{j : \beta_j^* \neq 0\}$ . For the folded concave penalized least square problem, the oracle solution has an explicit form of  $\widehat{\boldsymbol{\beta}}_{LS}^{oracle} = (\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle}, \mathbf{0})$  with

$$\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle} = (\mathbf{X}'_{\mathcal{A}} \mathbf{X}_{\mathcal{A}})^{-1} \mathbf{X}'_{\mathcal{A}} \mathbf{y},$$

and the Hessian matrix is  $\mathbf{X}'\mathbf{X}$  regardless of  $\boldsymbol{\beta}$ . Applying Theorems 1 and 2, we can derive the following theorem with explicit upper bounds for  $\delta_1$  and  $\delta_2$ , which depends only on behavior of the oracle estimator.

**Theorem 3.** Let  $\delta_0^{LS} = \Pr \left( \|\widehat{\boldsymbol{\beta}}_{LS}^{initial} - \boldsymbol{\beta}^*\|_{\max} > a_0 \lambda \right)$ . Suppose that

(A1)  $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \varepsilon$  with  $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n)$  being i.i.d. sub-Gaussian( $\sigma$ ) for some fixed constant  $\sigma > 0$ , i.e.  $E[\exp(t\varepsilon_i^2)] \leq \exp(\sigma^2 t^2/2)$ .

The LLA algorithm initiated by  $\widehat{\boldsymbol{\beta}}_{LS}^{initial}$  converges to the oracle estimator  $\widehat{\boldsymbol{\beta}}_{LS}^{oracle}$  after two iterations with a probability at least  $1 - \delta_0^{LS} - \delta_1^{LS} - \delta_2^{LS}$ , where

$$\delta_1^{LS} = 2(p - s) \cdot \exp \left( -\frac{a_1^2 n \lambda^2}{2M\sigma^2} \right)$$

and

$$\delta_2^{LS} = 2s \cdot \exp \left( -\frac{n\lambda_{\min}}{2\sigma^2} (\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)^2 \right),$$

where  $\lambda_{\min} = \lambda_{\min}(\frac{1}{n} \mathbf{X}'_{\mathcal{A}} \mathbf{X}_{\mathcal{A}})$  and  $M = \max_j \frac{1}{n} \|\mathbf{x}_{(j)}\|_{\ell_2}^2$ , which is usually 1 due to normalization, with  $\mathbf{x}_{(j)} = (x_{1j}, \dots, x_{nj})'$ .

By Theorem 3 both tail probabilities  $\delta_1^{LS}$  and  $\delta_2^{LS}$  go to zero very quickly. Then it remains to bound  $\delta_0^{LS}$ . To analyze  $\delta_0^{LS}$  we should decide the initial estimator. Here we consider the LASSO estimator (Tibshirani, 1996) as a natural choice to initialize the LLA algorithm, where the LASSO estimator is defined by

$$\widehat{\boldsymbol{\beta}}_{LS}^{lasso} = \arg \min_{\boldsymbol{\beta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_{\ell_2}^2 + \lambda_{lasso} \|\boldsymbol{\beta}\|_{\ell_1}. \quad (5)$$

Note that LASSO corresponds to the LLA estimator with initial estimator  $\hat{\beta}^{initial} = \mathbf{0}$ . In order to derive the estimation bound on  $\hat{\beta}_{LS}^{lasso} - \beta^*$ , we invoke the following restricted eigenvalue condition,

$$(C1) \quad \kappa_{LS} = \min_{\mathbf{u} \neq 0: \|\mathbf{u}_{\mathcal{A}^c}\|_{\ell_1} \leq 3\|\mathbf{u}_{\mathcal{A}}\|_{\ell_1}} \frac{\|\mathbf{X}\mathbf{u}\|_{\ell_2}}{\sqrt{n}\|\mathbf{u}\|_{\ell_2}} \in (0, \infty).$$

Such a condition has been studied in Bickel et al. (2009); Van De Geer and Bühlmann (2009); Raskutti et al. (2010) and Negahban et al. (2012). Under the sub-Gaussian noise assumption (A1) and also the restricted eigenvalue condition (C1), the LASSO estimator can yield a unique optimal solution  $\hat{\beta}_{LS}^{lasso}$  such that

$$\|\hat{\beta}_{LS}^{lasso} - \beta^*\|_{\ell_2} \leq \frac{2}{\kappa_{LS}} s^{1/2} \lambda_{lasso}$$

with probability at least  $1 - c_2 \exp(-c_1 n \lambda_{lasso}^2)$  where  $c_1$  and  $c_2$  are two fixed positive constants. See Corollary 2 of Negahban et al. (2012) for more details. Thus, using this as upper bound for  $\|\hat{\beta}_{LS}^{lasso} - \beta^*\|_{\max}$ , it is easy for us to obtain the following corollary.

**Corollary 3.** *Under the assumptions of (A0), (A1) and (C1), as long as  $\lambda$  is chosen to be greater than  $2(a_0 \kappa_{LS})^{-1} s^{1/2} \lambda_{lasso}$ , the LLA algorithm initiated by  $\hat{\beta}_{LS}^{lasso}$  converges to the oracle estimator  $\hat{\beta}_{LS}^{oracle}$  after two iterations with a probability at least  $1 - c_2 \exp(-c_1 n \lambda_{lasso}^2) - \delta_1^{LS} - \delta_2^{LS}$ , where  $\delta_1^{LS}$  and  $\delta_2^{LS}$  are given in Theorem 3.*

**Remark 4.** Before concluding this example we would like to emphasize that Theorem 3 is independent of the initial estimator. Although we have considered using the LASSO penalized least squares estimator as the initial estimator, we can also use Dantzig selector (Candes and Tao, 2007) as the initial estimator and the same analysis can go through under condition (C1) (Bickel et al., 2009).

### 3.2 Sparse logistic regression

The second example is the folded concave penalized logistic regression. Assume that

(A2) the conditional distribution of  $y_i$  given  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ ) is a Bernoulli distribution with  $\Pr(y_i = 1 | \mathbf{x}_i, \beta^*) = \exp(\mathbf{x}_i' \beta^*) / (1 + \exp(\mathbf{x}_i' \beta^*))$ .

Then, the penalized logistic regression is given by

$$\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_i \{-y_i \mathbf{x}'_i \boldsymbol{\beta} + \psi(\mathbf{x}'_i \boldsymbol{\beta})\} + \sum_j P_\lambda(|\beta_j|), \quad (6)$$

where  $\psi(t) = \log(1 + \exp(t))$  is the canonical link function. This model is the canonical statistical model for high-dimensional binary classification problems, and it is a classical example of the generalized linear model.

The oracle estimator is given by

$$\hat{\boldsymbol{\beta}}_{Logit}^{oracle} = (\hat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle}, \mathbf{0}) = \arg \min_{\boldsymbol{\beta}: \boldsymbol{\beta}_{\mathcal{A}^c} = \mathbf{0}} \frac{1}{n} \sum_i \{-y_i \mathbf{x}'_i \boldsymbol{\beta} + \psi(\mathbf{x}'_i \boldsymbol{\beta})\}.$$

For ease of presentation, we define

$$\boldsymbol{\mu}(\boldsymbol{\beta}) = (\psi'(\mathbf{x}'_1 \boldsymbol{\beta}), \dots, \psi'(\mathbf{x}'_n \boldsymbol{\beta}))'$$

and

$$\boldsymbol{\Sigma}(\boldsymbol{\beta}) = \text{diag}\{\psi''(\mathbf{x}'_1 \boldsymbol{\beta}), \dots, \psi''(\mathbf{x}'_n \boldsymbol{\beta})\}.$$

In addition, we introduce the following three useful quantities:

$$\begin{aligned} Q_1 &= \max_j \lambda_{\max} \left( \frac{1}{n} \mathbf{X}'_{\mathcal{A}} \text{diag}\{|\mathbf{x}_{(j)}|\} \mathbf{X}_{\mathcal{A}} \right); \\ Q_2 &= \left\| \left( \frac{1}{n} \mathbf{X}'_{\mathcal{A}} \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}} \right)^{-1} \right\|_{\ell_\infty}; \\ Q_3 &= \left\| \mathbf{X}'_{\mathcal{A}^c} \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}} (\mathbf{X}'_{\mathcal{A}} \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}})^{-1} \right\|_{\ell_\infty}. \end{aligned}$$

in which  $\text{diag}\{|\mathbf{x}_{(j)}|\}$  is a diagonal matrix with elements  $\{|x_{ij}|\}_{i=1}^n$ .

We first derive bounds on  $\delta_1$  and  $\delta_2$ .

**Theorem 4.** *Let  $\delta_0^{Logit} = \Pr \left( \|\hat{\boldsymbol{\beta}}_{Logit}^{initial} - \boldsymbol{\beta}^*\|_{\max} > a_0 \lambda \right)$ . Under Assumption (A2), the LLA algorithm initiated by  $\hat{\boldsymbol{\beta}}_{Logit}^{initial}$  converges to the oracle estimator  $\hat{\boldsymbol{\beta}}_{Logit}^{oracle}$  after two iterations with a probability at least  $1 - \delta_0^{Logit} - \delta_1^{Logit} - \delta_2^{Logit}$ , where*

$$\begin{aligned} \delta_1^{Logit} &= 2s \cdot \exp \left( -\frac{n}{M} \cdot \min \left\{ \frac{2}{Q_1^2 Q_2^4 s^2}, \frac{a_1^2 \lambda^2}{2(1 + 2Q_3)^2} \right\} \right) \\ &\quad + 2(p - s) \cdot \exp \left( -\frac{a_1^2 n \lambda^2}{2M} \right), \end{aligned}$$

where  $M = \max_j n^{-1} \|\mathbf{x}_{(j)}\|_{\ell_2}^2$  and

$$\delta_2^{Logit} = 2s \cdot \exp \left( -\frac{n}{M Q_2^2} \cdot \min \left\{ \frac{2}{Q_1^2 Q_2^2 s^2}, \frac{1}{2} (\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)^2 \right\} \right).$$

Under fairly weak assumptions, both  $\delta_1^{Logit}$  and  $\delta_2^{Logit}$  go to zero very quickly. The remaining challenge is to bound  $\delta_0^{Logit}$ . We consider using the  $\ell_1$ -penalized maximum likelihood estimator as the initial estimator, i.e.

$$\hat{\beta}_{Logit}^{lasso} = \arg \min_{\beta} \frac{1}{n} \sum_i \{-y_i \mathbf{x}'_i \beta + \psi(\mathbf{x}'_i \beta)\} + \lambda_{lasso} \|\beta\|_{\ell_1}.$$

In the following theorem, we provide the estimation bound on  $\hat{\beta}_{Logit}^{lasso} - \beta^*$ .

**Theorem 5.** *Let  $m = \max_{(i,j)} |x_{ij}|$ . Suppose that*

$$(C2) \quad \kappa_{Logit} = \min_{\mathbf{u} \neq \mathbf{0}: \|\mathbf{u}_{\mathcal{A}^c}\|_{\ell_1} \leq 3\|\mathbf{u}_{\mathcal{A}}\|_{\ell_1}} \frac{\mathbf{u}' \nabla^2 \ell_n^{Logit}(\beta) \mathbf{u}}{\mathbf{u}' \mathbf{u}} \in (0, \infty).$$

*Then the LASSO estimator  $\hat{\beta}_{Logit}^{lasso}$  with  $\lambda_{lasso} \leq \kappa_{Logit} (20ms)^{-1}$  satisfies*

$$\|\hat{\beta}_{Logit}^{lasso} - \beta^*\|_{\ell_2} \leq 5\kappa_{Logit}^{-1} s^{1/2} \lambda_{lasso}$$

*with a probability at least*

$$1 - 2p \cdot \exp\left(-\frac{1}{2M} n \lambda_{lasso}^2\right)$$

In light of Theorem 5, we can obtain the following corollary.

**Corollary 4.** *Under the assumptions of (A0), (A2) and (C2), as long as  $\lambda$  is chosen to be greater than  $5(a_0 \kappa_{LS})^{-1} s^{1/2} \lambda_{lasso}$ , the LLA algorithm initiated by  $\hat{\beta}_{Logit}^{lasso}$  converges to the oracle estimator  $\hat{\beta}_{Logit}^{oracle}$  after two iterations with a probability at least  $1 - 2p \exp(-\frac{1}{2M} n \lambda_{lasso}^2) - \delta_1^{Logit} - \delta_2^{Logit}$ , where  $\delta_1^{Logit}$  and  $\delta_2^{Logit}$  are given in Theorem 4.*

### 3.3 Sparse precision matrix estimation

The third example is the folded concave penalized Gaussian quasi-likelihood estimator for the sparse precision matrix estimation problem, i.e.

$$\min_{\Theta \succ 0} -\log \det(\Theta) + \langle \Theta, \hat{\Sigma}_n \rangle + \sum_{(j,k): j \neq k} P_{\lambda}(|\theta_{jk}|), \quad (7)$$

where  $\hat{\Sigma}_n = (\hat{\sigma}_{ij}^n)_{q \times q}$  is the sample covariance matrix estimator. In particular, under the assumption of the Gaussian distribution, the sparse precision matrix can be interpreted as a sparse Gaussian graphical model (Meinshausen and Bühlmann, 2006; Yuan and Lin, 2007;

Lam and Fan, 2009). In this example the target “parameter”  $\boldsymbol{\beta}^*$  is the true precision matrix  $\boldsymbol{\Theta}^* = (\theta_{jk}^*)_{q \times q}$ , and the corresponding support set is  $\mathcal{A} = \{(j, k) : \theta_{jk}^* \neq 0\}$ . Due to the symmetric structure of  $\boldsymbol{\Theta}^*$ , the dimension is  $p = q(q+1)/2$  and the cardinality of  $\mathcal{A}$  is  $s = \#\{(j, k) : j \leq k \text{ \& } \theta_{jk}^* \neq 0\}$ .

Now we introduce the oracle precision matrix estimator as follows,

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_G^{oracle} &= \arg \min_{\boldsymbol{\Theta} \succ 0} -\log \det(\boldsymbol{\Theta}) + \langle \boldsymbol{\Theta}, \hat{\boldsymbol{\Sigma}}_n \rangle \\ &\text{subject to } \theta_{jk} = 0, \forall (j, k) \in \mathcal{A}^c. \end{aligned}$$

Then we can write  $\hat{\boldsymbol{\Theta}}_G^{oracle} = (\hat{\boldsymbol{\Theta}}_{\mathcal{A}}^{oracle}, \hat{\boldsymbol{\Theta}}_{\mathcal{A}^c}^{oracle}) = (\hat{\boldsymbol{\Theta}}_{\mathcal{A}}^{oracle}, \mathbf{0})$ . For ease of notation, we define  $\hat{\boldsymbol{\Sigma}}_G^{oracle} = (\hat{\boldsymbol{\Theta}}_G^{oracle})^{-1}$ . Similarly, we partition  $\hat{\boldsymbol{\Sigma}}_n$  and  $\hat{\boldsymbol{\Sigma}}_G^{oracle}$  in terms of  $\mathcal{A}$ , i.e.  $\hat{\boldsymbol{\Sigma}}_n = (\hat{\boldsymbol{\Sigma}}_{\mathcal{A}}^n, \hat{\boldsymbol{\Sigma}}_{\mathcal{A}^c}^n)$  and  $\hat{\boldsymbol{\Sigma}}_G^{oracle} = (\hat{\boldsymbol{\Sigma}}_{\mathcal{A}}^{oracle}, \hat{\boldsymbol{\Sigma}}_{\mathcal{A}^c}^{oracle})$ . Note that the Hessian matrix of the negative log-quasi-likelihood function has the explicit expression of  $\mathbf{H}^* = (\boldsymbol{\Theta}^*)^{-1} \otimes (\boldsymbol{\Theta}^*)^{-1} = \boldsymbol{\Sigma}^* \otimes \boldsymbol{\Sigma}^*$ . We also define

$$K_1 = \|\boldsymbol{\Sigma}^*\|_{\ell_\infty}, \quad K_2 = \|(\mathbf{H}_{\mathcal{A}\mathcal{A}}^*)^{-1}\|_{\ell_\infty}, \quad \text{and} \quad K_3 = \|\mathbf{H}_{\mathcal{A}^c\mathcal{A}}^*(\mathbf{H}_{\mathcal{A}\mathcal{A}}^*)^{-1}\|_{\ell_\infty}.$$

We also define the maximal degree as  $d = \max_j \#\{k : \theta_{jk}^* \neq 0\}$ .

In the next theorem, we derive explicit bounds for  $\delta_1$  and  $\delta_2$ . For space consideration, we only consider the Gaussian distribution. Indeed, we can obtain exactly the same convergence result of the LLA algorithm for the folded concave penalized Gaussian quasi-likelihood problem under the exponential tail or the polynomial tail condition as in Cai et al. (2011).

To bound  $\delta_1$  and  $\delta_2$  under the Gaussian assumption, we cite a large deviation result by Saulis and Statulevicius (1991) and Bickel and Levina (2008): for any  $\nu$  such that  $|\nu| \leq \delta$ ,

$$\Pr(|\hat{\sigma}_{ij}^n - \sigma_{ij}^*| \geq \nu) \leq C_0 \exp(-c_0 n \nu^2) \quad (8)$$

where  $\delta$ ,  $c_0$  and  $C_0$  depend on  $\max_i \sigma_{ii}^*$  only.

**Theorem 6.** Let  $\delta_0^G = \Pr\left(\|\hat{\boldsymbol{\Theta}}_G^{initial} - \boldsymbol{\Theta}^*\|_{\max} > a_0 \lambda\right)$ . Suppose that

$$(A0') \quad \|\boldsymbol{\Theta}_{\mathcal{A}}^*\|_{\min} > (a+1)\lambda.$$

and we further assume that

$$(A3) \quad \mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)' \text{ are i.i.d. Gaussian random samples with the covariance matrix } \boldsymbol{\Sigma}^*.$$

The LLA algorithm initiated by  $\hat{\Theta}_G^{initial}$  converges to the oracle estimator  $\hat{\Theta}_G^{oracle}$  after two iterations with a probability at least  $1 - \delta_0^G - \delta_1^G - \delta_2^G$ , where

$$\begin{aligned} \delta_1^G &= C_0 s \cdot \exp \left( -\frac{c_0}{4} n \cdot \min \left\{ \frac{a_1^2 \lambda^2}{(2K_3 + 1)^2}, \frac{1}{9K_1^2 K_2^2 d^2}, \frac{1}{9K_1^6 K_2^4 d^2} \right\} \right) \\ &\quad + C_0 (p - s) \cdot \exp \left( -\frac{c_0 a_1^2}{4} n \lambda^2 \right) \end{aligned}$$

and

$$\delta_2^G = C_0 s \cdot \exp \left( -\frac{c_0 n}{4K_2^2} \cdot \min \left\{ \frac{1}{9K_1^2 d^2}, \frac{1}{9K_1^6 K_2^2 d^2}, (\|\Theta_{\mathcal{A}}^*\|_{\min} - a\lambda)^2 \right\} \right).$$

Theorem 6 tells us that both  $\delta_1^G$  and  $\delta_2^G$  go to zero very quickly. Now we only need to deal with  $\delta_0^G$ . To initialize the LLA algorithm, we consider using the constrained  $\ell_1$  minimization estimator (CLIME) by Cai et al. (2011), i.e.

$$\hat{\Theta}_G^{clime} = \arg \min_{\Theta} \|\Theta\|_1 \quad \text{subject to } \|\hat{\Sigma}_n \Theta - \mathbf{I}\|_{\max} \leq \lambda_{clime}.$$

To obtain the convergence rate of  $\hat{\Theta}_G^{clime}$ , we write  $\|\Theta^*\|_{\ell_1} = L$ . As discussed in Cai et al. (2011), it is reasonable to assume that  $L$  is upper bounded by a constant or  $L$  is some slowly diverging quantity, because  $\Theta^*$  has a few nonzero entries in each row. We combine the concentration bound (8) and the same line of proof as in Cai et al. (2011) to show that

$$\|\hat{\Theta}_G^{clime} - \Theta^*\|_{\max} \leq 4L\lambda_{clime}$$

with a probability at least  $1 - C_0 p \cdot \exp(-\frac{c_0}{L^2} n \lambda_{clime}^2)$ .

Thus we have the following corollary.

**Corollary 5.** *Under the assumptions of (A0') and (A3), as long as  $\lambda$  is chosen to be greater than  $4a_0^{-1} L \lambda_{clime}$ , the LLA algorithm initiated by  $\hat{\Theta}_G^{clime}$  converges to the oracle estimator  $\hat{\Theta}_G^{oracle}$  after two iterations with a probability at least  $1 - C_0 p \cdot \exp(-\frac{c_0}{L^2} n \lambda_{clime}^2) - \delta_1^G - \delta_2^G$ .*

### 3.4 Comments on the Irrepresentable Condition

So far we have demonstrated the applications of Theorems 1–2 for the LLA algorithm on three classical sparse estimation problems. It is well-known that the irrepresentable condition is necessary for the  $\ell_1$  penalization method to have the selection consistency property. Here we list the corresponding irrepresentable condition for the  $\ell_1$  penalized least squares, the  $\ell_1$  penalized logistic regression and the  $\ell_1$  penalized precision matrix estimation (Zhao and Yu, 2006; Ravikumar et al., 2008; Wainwright, 2009; Ravikumar et al., 2010):

(C3) the  $\ell_1$  penalized least squares: there exists some positive constant  $\gamma_{LS} \in (0, 1)$  such that

$$\|\mathbf{X}'_{\mathcal{A}^c} \mathbf{X}_{\mathcal{A}} (\mathbf{X}'_{\mathcal{A}} \mathbf{X}_{\mathcal{A}})^{-1}\|_{\ell_{\infty}} \leq 1 - \gamma_{LS};$$

(C4) the  $\ell_1$  penalized logistic regression: there exists some positive constant  $\gamma_{Logit} \in (0, 1)$  such that

$$\|\mathbf{X}'_{\mathcal{A}^c} \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}} (\mathbf{X}'_{\mathcal{A}} \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}})^{-1}\|_{\ell_{\infty}} \leq 1 - \gamma_{Logit};$$

(C5) the  $\ell_1$  penalized precision matrix estimation: there exists some positive constant  $\gamma_G \in (0, 1)$  such that

$$\|\mathbf{H}^*_{\mathcal{A}^c \mathcal{A}} (\mathbf{H}^*_{\mathcal{A} \mathcal{A}})^{-1}\|_{\ell_{\infty}} \leq 1 - \gamma_G.$$

All these conditions have been argued to be too restrictive, for example, see Zou (2006); Zhang (2010a); Fan and Lv (2011); Cai et al. (2011) and Xue et al. (2012). From our analysis it is clear that our theory does not require the initial estimator to be selection consistent. Thus we do not need to use these irrepresentable conditions even when the  $\ell_1$  penalized estimator is used as the initial estimator. This message is the most interesting in the case of sparse precision matrix estimation. We propose to use the CLIME as the initial estimator in the LLA algorithm. The reason is that a nice bound can be established for the CLIME under the elementwise maximum norm which is exactly what we need in order to apply Theorems 1 and 2. It is also interesting to see that although the sparse precision matrix estimation is the most complicated one among three examples, it actually requires the weakest regularity conditions to apply Theorems 1 and 2. We have used the restricted eigenvalue conditions (C1) and (C2) for sparse least squares regression and sparse logistic regression. Based on the current literature, it seems very difficult, if not impossible, to greatly relax (C1) and (C2) while keeping a nice bound on the estimation accuracy of the  $\ell_1$  penalized least squares/logistic regression estimator under the  $\ell_2$  loss. According to Bickel et al. (2009), the restricted eigenvalue condition (C1) is also used to derive estimation bounds for the Dantzig selector. Hence this condition is still needed if we use the Dantzig selector instead of the LASSO as the initial estimator. In contrast, in the sparse precision matrix estimation problem we do not need to impose any structure assumption on  $\boldsymbol{\Sigma}^*$  or the Hessian matrix  $\mathbf{H}^* = \boldsymbol{\Sigma}^* \otimes \boldsymbol{\Sigma}^*$ . The condition on  $\|\boldsymbol{\Theta}^*\|_{\ell_1}$  is not strong under the strong sparsity assumption



on  $\Theta^*$ . From this perspective, the sparse precision matrix estimation example is the best among the three to demonstrate the power and application of Theorems 1 and 2.

## 4 Simulation Studies

In this section we use simulation to examine the finite sample properties of the folded concave penalized estimation for solving three classical problems, i.e., sparse linear regression, sparse logistic regression and sparse precision matrix estimation. We use several different local solution algorithms to compute SCAD/MCP penalized estimators. We fix  $a = 3.7$  in the SCAD and  $a = 2$  in the MCP as suggested in Fan and Li (2001) and Zhang (2010a) respectively. We also include the LASSO penalized estimator in the study.

### 4.1 Sparse linear regression and logistic regression models

First we simulated the independent random samples  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$  from the following four sparse linear regression and logistic regression models.

Models 1 and 2 are sparse linear models.

**Model 1:**  $y = \mathbf{x}'\boldsymbol{\beta}^* + \varepsilon$  where  $\boldsymbol{\beta}^* = (3, 1.5, 0, 0, 2, 0_{p-5})$ ,  $\varepsilon \sim N(0, 1)$  and  $\mathbf{x} \sim N_p(0, \boldsymbol{\Sigma})$  with  $\boldsymbol{\Sigma} = (0.5^{|i-j|})_{p \times p}$ .

**Model 2:** The setup is the same as in Model 1, except that  $\boldsymbol{\beta}^*$  is constructed by randomly choosing 10 elements in  $\boldsymbol{\beta}^*$  as independent Bernoulli random samples with equal probability to be 1 or  $-1$ , and setting the other  $p - 10$  elements as zeros.

We let  $n = 100$  and  $p = 500$  &  $1000$ . We also generated an independent validation set of sample size 100 to tune each estimator. The validation error of a generic estimator  $\hat{\boldsymbol{\beta}}$  is defined as  $\sum_{i \in \text{validation}} (y_i - \mathbf{x}_i' \hat{\boldsymbol{\beta}})^2$ .

Models 3 and 4 are sparse logistic regression models.

**Model 3:**  $y$  follows a Bernoulli distribution with the probability of success being  $\exp(\mathbf{x}'\boldsymbol{\beta}^*) / (1 + \exp(\mathbf{x}'\boldsymbol{\beta}^*))$ , where  $\boldsymbol{\beta}^* = (3, 1.5, 0, 0, 2, 0_{p-5})$  and  $\mathbf{x} \sim N_p(0, \boldsymbol{\Sigma})$  with  $\boldsymbol{\Sigma} = (0.5^{|i-j|})_{p \times p}$ .

**Model 4:** The setup is the same as in Model 3, except that  $\beta^*$  is constructed by randomly choosing 10 elements in  $\beta^*$  as  $t_1 s_1, \dots, t_{10} s_{10}$  and setting the other  $p - 10$  elements as zeros, where  $t_j$ 's are independently drawn from  $\text{Unif}(1, 2)$ , and  $s_j$ 's are independent Bernoulli random samples with  $\Pr(s_j = 1) = \Pr(s_j = -1) = 0.5$ .

We let  $n = 200$  and  $p = 500$  &  $1000$ . We also generated an independent validation set of sample size 200 to tune each estimator. The validation error of a generic estimator  $\hat{\beta}$  is defined as

$$\sum_{i \in \text{validation}} \left( -y_i \mathbf{x}_i' \hat{\beta} + \log(1 + \exp(\mathbf{x}_i' \hat{\beta})) \right).$$

We computed the LASSO penalized linear/logistic regression by the popular R package *glmnet* (Friedman et al., 2012) and chose its penalization parameter by minimizing the validation error. We implemented three local solutions of SCAD/MCP. The first local solution was computed by using coordinate descent. We denote it by SCAD-cd/MCP-cd. The second local solution, denoted by SCAD-lla0/MCP-lla0, was computed by the LLA algorithm with zeros as its initial estimator. The third local solution, denoted by SCAD-lla\*/MCP-lla\*, was computed by the LLA algorithm with the tuned LASSO estimator as its initial estimator. SCAD-lla\*/MCP-lla\* was designed according to the theoretical analysis in Sections 3.1 and 3.2. Given an initial estimator, we implemented the LLA algorithm for SCAD/MCP by using *glmnet* to solve the weighted  $\ell_1$  penalized estimator at each LLA step. For each local solution of SCAD/MCP its penalization parameter was chosen by minimizing the validation error.

---

---

Tables 1–4 are about here.

---

---

For each model, we generated 100 independent datasets, each consisting  $n$  training samples and  $n$  validation samples. Estimation accuracy is measured by the average  $\ell_1$  loss  $\|\hat{\beta} - \beta^*\|_{\ell_1}$  over the 100 replications, and selection accuracy is evaluated by the average counts of false positive and false negative over the 100 replications. The simulation results of Models 1–4 are summarized in Tables 1–4, respectively. Needless to say, all SCAD/MCP solutions perform much better than the LASSO estimator. This is a familiar message from previous works on folded concave penalized estimation (Fan and Li, 2001; Zhang, 2010a; Fan and Lv, 2011). We would like to emphasize on comparison between local solutions of

SCAD/MCP. First, it is very interesting to see that the local solutions of SCAD/MCP are very different. Even the two LLA local solutions are noticeably different. This clearly suggest that the unique minimizer argument does not apply here. Second, SCAD-lla<sup>\*</sup> and MCP-lla<sup>\*</sup> achieve the best performance in both estimation and selection, which gives numeric evidence to the theoretical analysis in Section 3.1 and 3.2. When the average FP and FN are zero, the estimator is model selection consistent and is also an evidence of finding the oracle estimator.

## 4.2 Sparse Gaussian graphical models

We simulated  $n$  independent random vector from  $N_q(\mathbf{0}, \mathbf{\Sigma}^*)$  with a sparse precision matrix  $\mathbf{\Theta}^* = (\mathbf{\Sigma}^*)^{-1}$ . Models 5 and 6 consider two different sparsity patterns of  $\mathbf{\Theta}^*$ .

**Model 5:**  $\mathbf{\Theta}^*$  is a tridiagonal matrix by constructing  $\mathbf{\Sigma}^* = (\sigma_{ij}^*)_{q \times q}$  as an AR(1) covariance matrix with  $\sigma_{ij}^* = \exp(-|s_i - s_j|)$  for  $s_1 < \dots < s_q$  which are constructed by simulating  $s_q - s_{q-1}, s_{q-1} - s_{q-2}, \dots, s_2 - s_1$  independently from  $\text{Unif}(0.5, 1)$ ;

**Model 6:**  $\mathbf{\Theta}^* = \mathbf{U}'_{q \times q} \mathbf{U}_{q \times q} + \mathbf{I}_{q \times q}$  where  $\mathbf{U} = (u_{ij})_{q \times q}$  has zero diagonals and exactly 100 nonzero off-diagonal entries. The nonzero entries are generated by  $u_{ij} = t_{ij}s_{ij}$  where  $t_{ij}$ 's are independently drawn from  $\text{Unif}(1, 2)$ , and  $s_{ij}$ 's are independent Bernoulli random variables with  $\Pr(s_{ij} = 1) = \Pr(s_{ij} = -1) = 0.5$ .

We also generated an independent validation set of sample size  $n$  to tune each estimator. The validation error of a generic estimator  $\hat{\mathbf{\Theta}}$  is defined as  $-\log \det(\hat{\mathbf{\Theta}}) + \langle \hat{\mathbf{\Theta}}, \hat{\mathbf{\Sigma}}_n^{\text{validation}} \rangle$ . In our simulation we let  $q = 100$  and  $n = 100$  &  $200$ .

We computed the  $\ell_1$  penalized Gaussian likelihood estimator, denoted by GLASSO, by using the popular R package *glmnet* (Friedman et al., 2011). For ease of presentation, we use GSCAD/GMCP to denote the SCAD/MCP penalized Gaussian likelihood estimator. We computed the CLIME by the R package *clime* (Cai et al., 2012). GLASSO and CLIME were tuned by minimizing its validation error. We considered two LLA local solutions of GSCAD/GMCP. The first one, denoted by GSCAD-lla0/GMCP-lla0, uses  $\text{diag}(\hat{\Sigma}_{jj}^{-1})$  as the initial estimator in the LLA algorithm. The second one, denoted by GSCAD-lla<sup>\*</sup>/GMCP-lla<sup>\*</sup>, uses the tuned CLIME as the initial estimator in the LLA algorithm. GSCAD-lla<sup>\*</sup>/GMCP-lla<sup>\*</sup> was designed according to the theoretical analysis in Section 3.3. In

the LLA algorithm for GSCAD/GMCP we used *glasso* to compute the weighted  $\ell_1$  penalized Gaussian likelihood estimator at each LLA step. For each local solution of GSCAD/GMCP its penalization parameter was chosen by minimizing the validation error.

---



---

Tables 5–6 are about here.

---



---

For each model, we generated 100 independent datasets, each consisting  $n$  training samples and  $n$  validation samples. Estimation accuracy is measured by the average Operator norm loss  $\|\hat{\Theta} - \Theta\|_{\ell_2}$  and the average Frobenius norm loss  $\|\hat{\Theta} - \Theta\|_F$  over the 100 replications, and selection accuracy is evaluated by the average counts of false positive and false negative over the 100 replications. The simulation results are summarized in Tables 5 and 6. Again, we see that the two local solutions of GSCAD/GMCP are very different, which implies that the unique minimizer argument is invalid here. GSCAD-lla $^*$  and GMCP-lla $^*$  achieve the best finite sample performance in both estimation and selection, which gives numeric evidence to the theoretical analysis in Section 3.3.

## 5 Technical Proofs

### 5.1 Proof of Theorem 1

*Proof.* To simplify notation, we let  $\hat{\beta}^{(0)} = \hat{\beta}^{(initial)}$ . Under the event  $\{\|\hat{\beta}^{(0)} - \beta^*\|_{\max} \leq a_0\lambda\}$ , due to the assumption (A0), we have for  $j \in \mathcal{A}^c$

$$|\hat{\beta}_j^{(0)}| \leq \|\hat{\beta}^{(0)} - \beta^*\|_{\max} \leq a_0\lambda \leq a_2\lambda$$

and for  $j \in \mathcal{A}$

$$|\hat{\beta}_j^{(0)}| \geq \|\beta_{\mathcal{A}}^*\|_{\min} - \|\hat{\beta}^{(0)} - \beta^*\|_{\max} > a\lambda.$$

Thus by property (iv) of  $P_\lambda(\cdot)$ ,  $P'_\lambda(|\hat{\beta}_j^{(0)}|) = 0$  for all  $j \in \mathcal{A}$ . Hence,  $\hat{\beta}^{(1)}$  is the solution of the following convex optimization problem

$$\hat{\beta}^{(1)} = \arg \min_{\beta} \ell_n(\beta) + \sum_{j \in \mathcal{A}^c} P'_\lambda(|\hat{\beta}_j^{(0)}|) \cdot |\beta_j|. \quad (9)$$

By property (ii) & (iii) of  $P_\lambda(\cdot)$ , we have  $P'_\lambda(|\hat{\beta}_j^{(0)}|) \geq a_1\lambda$  for any  $j \in \mathcal{A}^c$ .

We now show that  $\hat{\beta}^{oracle}$  is the unique global solution to (9) under the additional condition  $\{\|\nabla_{\mathcal{A}^c} \ell_n(\hat{\beta}^{oracle})\|_{\max} < a_1 \lambda\}$ , i.e.  $\hat{\beta}^{(1)} = \hat{\beta}^{oracle}$ . To see this, note that by convexity, we have

$$\begin{aligned} \ell_n(\beta) &\geq \ell_n(\hat{\beta}^{oracle}) + \sum_j \nabla_j \ell_n(\hat{\beta}^{oracle})(\beta_j - \hat{\beta}_j^{oracle}) \\ &= \ell_n(\hat{\beta}^{oracle}) + \sum_{j \in \mathcal{A}^c} \nabla_j \ell_n(\hat{\beta}^{oracle})(\beta_j - \hat{\beta}_j^{oracle}), \end{aligned} \quad (10)$$

where (3) was used in the last equality. By (10) and  $\hat{\beta}_{\mathcal{A}^c}^{oracle} = \mathbf{0}$ , we have that for any  $\beta$

$$\begin{aligned} &\{\ell_n(\beta) + \sum_{j \in \mathcal{A}^c} P'_\lambda(|\hat{\beta}_j^{(0)}|)|\beta_j|\} - \{\ell_n(\hat{\beta}^{oracle}) + \sum_{j \in \mathcal{A}^c} P'_\lambda(|\hat{\beta}_j^{(0)}|)|\hat{\beta}_j^{oracle}|\} \\ &\geq \sum_{j \in \mathcal{A}^c} \{P'_\lambda(|\hat{\beta}_j^{(0)}|) - \nabla_j \ell_n(\hat{\beta}^{oracle}) \cdot \text{sign}(\beta_j)\} \cdot |\beta_j| \\ &\geq \sum_{j \in \mathcal{A}^c} \{a_1 \lambda - \nabla_j \ell_n(\hat{\beta}^{oracle}) \cdot \text{sign}(\beta_j)\} \cdot |\beta_j| \\ &\geq 0. \end{aligned}$$

The strict inequality holds unless  $\beta_j = 0, \forall j \in \mathcal{A}^c$ . This together with the uniqueness of the solution to (2) concludes that  $\hat{\beta}^{oracle}$  is the unique solution to (9). Hence,  $\hat{\beta}^{(1)} = \hat{\beta}^{oracle}$ , which completes the proof of Theorem 1.  $\square$

## 5.2 Proof of Theorem 2

*Proof.* Given that the LLA algorithm finds  $\hat{\beta}^{oracle}$  at the current iteration, we denote  $\hat{\beta}$  as the solution to the convex optimization problem in the next iteration of the LLA algorithm. Using  $\hat{\beta}_{\mathcal{A}^c}^{oracle} = \mathbf{0}$  and  $P'_\lambda(|\hat{\beta}_j^{oracle}|) = 0$  for  $j \in \mathcal{A}$  under the event  $\{\|\hat{\beta}_{\mathcal{A}}^{oracle}\|_{\min} > a\lambda\}$ , we have

$$\hat{\beta} = \arg \min_{\beta} \ell_n(\beta) + \sum_{j \in \mathcal{A}^c} \gamma \cdot |\beta_j|, \quad (11)$$

where  $\gamma = P'_\lambda(0) \geq a_1 \lambda$ . This problem is very similar to (9). Following the same lines of the proof as in Theorem 1, it can easily be seen that under the additional condition  $\{\|\nabla_{\mathcal{A}^c} \ell_n(\hat{\beta}^{oracle})\|_{\max} < a_1 \lambda\}$ ,  $\hat{\beta}^{oracle}$  is the unique solution to (11). Hence the loop within the LLA algorithm stops, which completes the proof of Theorem 2.  $\square$

### 5.3 Proof of Theorem 3

*Proof.* Note it is sufficient to directly bound  $\delta_1$  and  $\delta_2$  for the least-squares problem. Let  $\mathbf{H}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}}(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}$ . Then,

$$\begin{aligned}\nabla_{\mathcal{A}^c}\ell_n(\widehat{\boldsymbol{\beta}}_{LS}^{oracle}) &= \frac{1}{n}\mathbf{X}'_{\mathcal{A}^c}(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}_{LS}^{oracle}) \\ &= \frac{1}{n}(\mathbf{X}'_{\mathcal{A}^c}\mathbf{y} - \mathbf{H}_{\mathcal{A}}\mathbf{y}) \\ &= \frac{1}{n}\mathbf{X}'_{\mathcal{A}^c}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\varepsilon,\end{aligned}$$

where we used  $\mathbf{y} = \mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}_{\mathcal{A}}^* + \varepsilon$  in the last equality. Thus, by the union bound and the Chernoff bound, we have

$$\begin{aligned}\delta_1 &= \Pr(\|\mathbf{X}'_{\mathcal{A}^c}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\varepsilon\|_{\max} > a_1 n \lambda) \\ &\leq \sum_{j \in \mathcal{A}^c} \Pr(\|\mathbf{x}'_{(j)}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\varepsilon\|_{\max} > a_1 n \lambda) \\ &\leq 2 \sum_{j \in \mathcal{A}^c} \exp\left(-\frac{a_1^2 n^2 \lambda^2}{2\sigma^2 \cdot \|\mathbf{x}'_{(j)}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\|_{\ell_2}^2}\right).\end{aligned}$$

Using the fact that

$$\|\mathbf{x}'_{(j)}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\|_{\ell_2}^2 = \mathbf{x}'_{(j)}(\mathbf{I}_{n \times n} - \mathbf{H}_{\mathcal{A}})\mathbf{x}_{(j)} \leq \|\mathbf{x}_{(j)}\|_{\ell_2}^2 \leq nM,$$

we conclude that

$$\delta_1 \leq 2(p-s) \exp\left(-\frac{a_1^2 n \lambda^2}{2M\sigma^2}\right).$$

We now derive an upper bound for  $\delta_2$  in the least-squares problem. Noticing that

$$\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle} = (\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\mathbf{y} = \boldsymbol{\beta}_{\mathcal{A}}^* + (\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\varepsilon.$$

we have

$$\|\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{oracle}\|_{\min} \geq \|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - \|(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\varepsilon\|_{\max}.$$

Thus,

$$\delta_2 \leq \Pr(\|(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\varepsilon\|_{\max} \geq \|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda). \quad (12)$$

It remains to derive an explicit probability upper bound for (12). To facilitate the notation, we define

$$(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}} = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_s)',$$

namely  $\mathbf{u}_j = \mathbf{X}_{\mathcal{A}}(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{e}_j$ , where  $\mathbf{e}_j$  is the unit vector with  $j^{th}$  element 1. It is obvious that

$$\|\mathbf{u}_j\|_{\ell_2}^2 = \mathbf{e}'_j(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{e}_j \leq (n\lambda_{\min})^{-1}.$$

By the union bound and the Markov bound again, we have

$$\begin{aligned} & \Pr \left( \|(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\varepsilon\|_{\max} \geq \|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda \right) \\ & \leq 2 \sum_{j=1}^s \exp \left( \frac{1}{2}\sigma^2\|\mathbf{u}_j\|_{\ell_2}^2 t^2 - (\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)t \right), \end{aligned}$$

where any  $t > 0$ . By using the Chernoff bound argument, we set  $t = \sigma^{-2}\|\mathbf{u}_j\|_{\ell_2}^{-2}(\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)$  to obtain

$$\begin{aligned} & \Pr \left( \|(\mathbf{X}'_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}'_{\mathcal{A}}\varepsilon\|_{\max} \geq \|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda \right) \\ & \leq 2 \sum_{j=1}^s \exp \left( -\frac{(\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)^2}{2\sigma^2\|\mathbf{u}_j\|_{\ell_2}^2} \right) \\ & \leq 2s \exp \left( -\frac{n\lambda_{\min}}{2\sigma^2}(\|\boldsymbol{\beta}_{\mathcal{A}}^*\|_{\min} - a\lambda)^2 \right). \end{aligned}$$

Thus, we complete the proof of Theorem 3.  $\square$

## 5.4 Proof of Theorem 4

*Proof.* A translation of (3) into our setting becomes

$$\mathbf{X}'_{\mathcal{A}}\boldsymbol{\mu}(\widehat{\boldsymbol{\beta}}_{Logit}^{oracle}) = \mathbf{X}'_{\mathcal{A}}\mathbf{y}. \quad (13)$$

We now use this to derive the upper bound for  $\delta_2$ .

Define a map  $F : \mathbb{B}(r) \subset \mathbb{R}^p \rightarrow \mathbb{R}^p$  satisfying

$$F(\boldsymbol{\Delta}) = ((F_{\mathcal{A}}(\boldsymbol{\Delta}_{\mathcal{A}}))', \mathbf{0}')'$$

with

$$F_{\mathcal{A}}(\boldsymbol{\Delta}_{\mathcal{A}}) = (\mathbf{X}'_{\mathcal{A}}\boldsymbol{\Sigma}(\boldsymbol{\beta}^*)\mathbf{X}_{\mathcal{A}})^{-1} \cdot \mathbf{X}'_{\mathcal{A}}(\mathbf{y} - \boldsymbol{\mu}(\boldsymbol{\beta}^* + \boldsymbol{\Delta})) + \boldsymbol{\Delta}_{\mathcal{A}}$$

and the convex compact set

$$\mathbb{B}(r) = \{\boldsymbol{\Delta} \in \mathbb{R}^p : \|\boldsymbol{\Delta}_{\mathcal{A}}\|_{\max} \leq r, \boldsymbol{\Delta}_{\mathcal{A}^c} = \mathbf{0}\}$$

with  $r = 2Q_2 \cdot \|\frac{1}{n}\mathbf{X}'_{\mathcal{A}}(\boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{y})\|_{\max}$ . Our aim is to show

$$F(\mathbb{B}(r)) \subset \mathbb{B}(r) \quad (14)$$

when

$$\|\frac{1}{n}\mathbf{X}'_{\mathcal{A}}(\boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{y})\|_{\max} \leq \frac{1}{Q_1 Q_2^2 s}. \quad (15)$$

If (14) holds, by the Brouwer's fixed point theorem, there always exists a fixed point  $\widehat{\boldsymbol{\Delta}} \in \mathbb{B}(r)$  such that  $F(\widehat{\boldsymbol{\Delta}}) = \widehat{\boldsymbol{\Delta}}$ . It immediately follows that

$$\mathbf{X}'_{\mathcal{A}}\mathbf{y} = \mathbf{X}'_{\mathcal{A}}\boldsymbol{\mu}(\boldsymbol{\beta}^* + \widehat{\boldsymbol{\Delta}}) \quad \text{and} \quad \widehat{\boldsymbol{\Delta}}_{\mathcal{A}^c} = \mathbf{0},$$

which further implies that  $\boldsymbol{\beta}^* + \widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\beta}}_{\text{Logit}}^{\text{oracle}}$  by the uniqueness of the solution to (13). Thus,

$$\|\widehat{\boldsymbol{\beta}}_{\text{Logit}}^{\text{oracle}} - \boldsymbol{\beta}^*\|_{\max} = \|\widehat{\boldsymbol{\Delta}}\|_{\max} \leq r. \quad (16)$$

If further

$$\|\frac{1}{n}\mathbf{X}'_{\mathcal{A}}(\boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{y})\|_{\max} \leq \frac{1}{2Q_2}(\|\boldsymbol{\beta}^*\|_{\min} - a\lambda),$$

then we have

$$r \leq \|\boldsymbol{\beta}^*\|_{\min} - a\lambda$$

and by (16)

$$\|\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{\text{oracle}}\|_{\min} \geq \|\boldsymbol{\beta}^*\|_{\min} - \|\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{\text{oracle}} - \boldsymbol{\beta}^*\|_{\max} \geq a\lambda.$$

Therefore, we have

$$\delta_2 \leq \Pr \left( \|\frac{1}{n}\mathbf{X}'_{\mathcal{A}}(\boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{y})\|_{\max} > \min \left\{ \frac{1}{Q_1 Q_2^2 s}, \frac{1}{2Q_2}(\|\boldsymbol{\beta}^*\|_{\min} - a\lambda) \right\} \right).$$

By combining the union bound and the Hoeffding's bound as in Proposition 4(a) of Fan and Lv (2011), we have

$$\delta_2 \leq 2s \cdot \exp \left( -\frac{n}{MQ_2^2} \cdot \min \left\{ \frac{2}{Q_1^2 Q_2^2 s^2}, \frac{1}{2}(\|\boldsymbol{\beta}^*\|_{\min} - a\lambda)^2 \right\} \right).$$

We now derive (14). By using its Taylor expansion around  $\boldsymbol{\Delta} = \mathbf{0}$ ,

$$\mathbf{X}'_{\mathcal{A}}\boldsymbol{\mu}(\boldsymbol{\beta}^* + \boldsymbol{\Delta}) = \mathbf{X}'_{\mathcal{A}}\boldsymbol{\mu}(\boldsymbol{\beta}^*) + \mathbf{X}'_{\mathcal{A}}\boldsymbol{\Sigma}(\boldsymbol{\beta}^*)\mathbf{X}\boldsymbol{\Delta} + \mathbf{R}_{\mathcal{A}}(\widetilde{\boldsymbol{\Delta}}),$$

where with  $\widetilde{\boldsymbol{\Delta}}$  being on the line segment joining  $\mathbf{0}$  and  $\boldsymbol{\Delta}$ ,

$$\mathbf{R}_{\mathcal{A}}(\widetilde{\boldsymbol{\Delta}}) = \mathbf{X}'_{\mathcal{A}} \left( \boldsymbol{\Sigma}(\boldsymbol{\beta}^* + \widetilde{\boldsymbol{\Delta}}) - \boldsymbol{\Sigma}(\boldsymbol{\beta}^*) \right) \mathbf{X}\boldsymbol{\Delta}.$$



Since  $\Delta_{\mathcal{A}^c} = \mathbf{0}$  by the definition of  $\mathbb{B}(r)$ , we have  $\mathbf{X}\Delta = \mathbf{X}_{\mathcal{A}}\Delta_{\mathcal{A}}$ . By the mean-value theorem, the entrywise maximum of  $\mathbf{R}_{\mathcal{A}}(\tilde{\Delta})$  can be bounded as

$$\|\mathbf{R}_{\mathcal{A}}(\tilde{\Delta})\|_{\max} \leq \max_j \Delta'_{\mathcal{A}} \mathbf{X}'_{\mathcal{A}} \text{diag}\{|\mathbf{x}_{(j)}| \circ |\boldsymbol{\mu}''(\bar{\boldsymbol{\beta}})|\} \mathbf{X}_{\mathcal{A}} \Delta_{\mathcal{A}}$$

for  $\bar{\boldsymbol{\beta}}$  being on the line segment joining  $\boldsymbol{\beta}^*$  and  $\boldsymbol{\beta}^* + \tilde{\Delta}$ . Using the simple fact that  $|\psi'''(t)| = \theta(t)(1 - \theta(t)) \cdot |2\theta(t) - 1| \leq \frac{1}{4}$  with  $\theta(t) = (1 + \exp(t))^{-1} \in (0, 1)$ , we have

$$\|\mathbf{R}_{\mathcal{A}}(\tilde{\Delta})\|_{\max} \leq \frac{n}{4} Q_1 \cdot \|\Delta_{\mathcal{A}}\|_{\ell_2}^2 \leq \frac{n}{4} Q_1 s r^2. \quad (17)$$

Noting that

$$\begin{aligned} F_{\mathcal{A}}(\Delta_{\mathcal{A}}) &= (\mathbf{X}'_{\mathcal{A}} \Sigma(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}})^{-1} \mathbf{X}'_{\mathcal{A}} (\mathbf{y} - \boldsymbol{\mu}(\boldsymbol{\beta}^*) + \boldsymbol{\mu}(\boldsymbol{\beta}^*) - \boldsymbol{\mu}(\boldsymbol{\beta}^* + \Delta)) + \Delta_{\mathcal{A}} \\ &= (\mathbf{X}'_{\mathcal{A}} \Sigma(\boldsymbol{\beta}^*) \mathbf{X}_{\mathcal{A}})^{-1} \cdot (\mathbf{X}'_{\mathcal{A}} (\mathbf{y} - \boldsymbol{\mu}(\boldsymbol{\beta}^*)) - \mathbf{R}_{\mathcal{A}}(\tilde{\Delta})), \end{aligned}$$

we then use the triangle inequality to obtain

$$\begin{aligned} \|F_{\mathcal{A}}(\Delta_{\mathcal{A}})\|_{\max} &= \|(\mathbf{X}'_{\mathcal{A}} \Sigma(\boldsymbol{\beta}) \mathbf{X}_{\mathcal{A}})^{-1} \cdot (\mathbf{X}'_{\mathcal{A}} \mathbf{y} - \mathbf{X}'_{\mathcal{A}} \boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{R}_{\mathcal{A}}(\tilde{\Delta}))\|_{\max} \\ &\leq Q_2 \cdot \left( \left\| \frac{1}{n} \mathbf{X}'_{\mathcal{A}} (\boldsymbol{\mu}(\boldsymbol{\beta}^*) - \mathbf{y}) \right\|_{\max} + \frac{1}{n} \|\mathbf{R}_{\mathcal{A}}(\tilde{\Delta})\|_{\max} \right). \end{aligned}$$

By using (17) and the definition of  $r$ , we have

$$\|F_{\mathcal{A}}(\Delta_{\mathcal{A}})\|_{\max} \leq \frac{r}{2} + \frac{1}{4} Q_1 Q_2 s r^2 \leq r.$$

This establishes (14).

Next we prove the upper bound for  $\delta_1$ . Recall that  $\hat{\Delta} = \hat{\boldsymbol{\beta}}_{\text{Logit}}^{\text{oracle}} - \boldsymbol{\beta}^*$ . Let

$$\ell_n^{\text{Logit}}(\hat{\boldsymbol{\beta}}) = \frac{1}{n} \sum_{i=1}^n \{-y_i \mathbf{x}'_i \boldsymbol{\beta} + \psi(\mathbf{x}'_i \boldsymbol{\beta})\}.$$

By a Taylor expansion,

$$\begin{aligned} \nabla \ell_n^{\text{Logit}}(\hat{\boldsymbol{\beta}}_{\text{Logit}}^{\text{oracle}}) &= \nabla \ell_n^{\text{Logit}}(\boldsymbol{\beta}^*) + \nabla^2 \ell_n^{\text{Logit}}(\boldsymbol{\beta}^*) \cdot \hat{\Delta} \\ &\quad + \left( \nabla^2 \ell_n^{\text{Logit}}(\tilde{\boldsymbol{\beta}}) - \nabla^2 \ell_n^{\text{Logit}}(\boldsymbol{\beta}^*) \right) \cdot \hat{\Delta}, \end{aligned} \quad (18)$$

where  $\tilde{\boldsymbol{\beta}}$  is on the line segment joining  $\hat{\boldsymbol{\beta}}_{\text{Logit}}^{\text{oracle}}$  and  $\boldsymbol{\beta}^*$ . Observe that the first and second derivatives of  $\ell_n^{\text{Logit}}(\boldsymbol{\beta})$  can be explicitly written as

$$\nabla \ell_n^{\text{Logit}}(\boldsymbol{\beta}) = \frac{1}{n} \mathbf{X}' (\boldsymbol{\mu}(\boldsymbol{\beta}) - \mathbf{y}) \quad \text{and} \quad \nabla^2 \ell_n^{\text{Logit}}(\boldsymbol{\beta}) = \frac{1}{n} \mathbf{X}' \Sigma(\boldsymbol{\beta}) \mathbf{X}. \quad (19)$$

Now we define

$$\begin{aligned}\mathbf{R}(\Delta) &= \left( \nabla^2 \ell_n^{Logit}(\tilde{\beta}) - \nabla^2 \ell_n^{Logit}(\beta^*) \right) \cdot \hat{\Delta} \\ &= \mathbf{X}'(\Sigma(\beta^* + \Delta) - \Sigma(\beta^*)) \mathbf{X} \hat{\Delta}.\end{aligned}$$

We also partition  $\mathbf{R}(\Delta)$  with respect to  $\mathcal{A}$ , i.e.  $\mathbf{R}(\Delta) = (\mathbf{R}'_{\mathcal{A}}(\Delta), \mathbf{R}'_{\mathcal{A}^c}(\Delta))'$ . Let  $\tilde{\Delta} = \tilde{\beta} - \beta^*$ . Then, using  $\hat{\Delta}_{\mathcal{A}^c} = \mathbf{0}$ , we have  $\mathbf{X} \hat{\Delta} = \mathbf{X}_{\mathcal{A}} \hat{\Delta}_{\mathcal{A}}$ . Substituting this into (18), we obtain

$$\nabla_{\mathcal{A}} \ell_n^{Logit}(\hat{\beta}_{Logit}^{oracle}) = \nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*) + \frac{1}{n} \mathbf{X}'_{\mathcal{A}} \Sigma(\beta) \mathbf{X}_{\mathcal{A}} \hat{\Delta}_{\mathcal{A}} + \frac{1}{n} \mathbf{R}_{\mathcal{A}}(\tilde{\Delta}), \quad (20)$$

and

$$\nabla_{\mathcal{A}^c} \ell_n^{Logit}(\hat{\beta}_{Logit}^{oracle}) = \nabla_{\mathcal{A}^c} \ell_n^{Logit}(\beta^*) + \frac{1}{n} \mathbf{X}'_{\mathcal{A}^c} \Sigma(\beta) \mathbf{X}_{\mathcal{A}} \hat{\Delta}_{\mathcal{A}} + \frac{1}{n} \mathbf{R}_{\mathcal{A}^c}(\tilde{\Delta}). \quad (21)$$

Using (19) and  $\nabla_{\mathcal{A}} \ell_n^{Logit}(\hat{\beta}_{Logit}^{oracle}) = \mathbf{0}$ , we can solve for  $\hat{\Delta}_{\mathcal{A}}$  from (20) and substitute it into (21) to obtain

$$\begin{aligned}& \nabla_{\mathcal{A}^c} \ell_n^{Logit}(\hat{\beta}_{Logit}^{oracle}) \\ &= \mathbf{X}'_{\mathcal{A}^c} \Sigma(\beta^*) \mathbf{X}_{\mathcal{A}} (\mathbf{X}'_{\mathcal{A}} \Sigma(\beta^*) \mathbf{X}_{\mathcal{A}})^{-1} \left( -\frac{1}{n} \mathbf{X}'_{\mathcal{A}} (\mu(\beta^*) - \mathbf{y}) - \frac{1}{n} \mathbf{R}_{\mathcal{A}}(\tilde{\Delta}) \right) \\ & \quad + \frac{1}{n} \mathbf{X}'_{\mathcal{A}^c} (\mu(\beta^*) - \mathbf{y}) + \frac{1}{n} \mathbf{R}_{\mathcal{A}^c}(\tilde{\Delta}).\end{aligned}$$

Recall that (16) under condition (15). If in addition under the event

$$\{\|\nabla_{\mathcal{A}^c} \ell_n^{Logit}(\beta^*)\|_{\max} < \frac{a_1 \lambda}{2}\} \cap \{\|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max} \leq \frac{a_1 \lambda}{4Q_3 + 2}\},$$

we can follow the same lines of proof as in (17) to show that

$$\|\mathbf{R}(\tilde{\Delta})\|_{\max} \leq \frac{n}{4} Q_1 \|\hat{\Delta}_{\mathcal{A}}\|_{\ell_2}^2 \leq \frac{n}{4} Q_1 s r^2,$$

where  $r = 2Q_2 \cdot \|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max}$ . Noticing that under condition (15)

$$\frac{n}{4} Q_1 s r^2 = s n Q_1 Q_2^2 \cdot \|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max}^2 \leq n \cdot \|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max},$$

under the same event we have

$$\begin{aligned}& \|\nabla_{\mathcal{A}^c} \ell_n^{Logit}(\hat{\beta}_{Logit}^{oracle})\| \\ & \leq Q_3 \cdot \left( \|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max} + \frac{1}{n} \|\mathbf{R}_{\mathcal{A}}(\tilde{\Delta})\|_{\max} \right) \\ & \quad + \|\nabla_{\mathcal{A}^c} \ell_n^{Logit}(\beta^*)\|_{\max} + \frac{1}{n} \|\mathbf{R}_{\mathcal{A}^c}(\tilde{\Delta})\|_{\max} \\ & \leq (2Q_3 + 1) \cdot \|\nabla_{\mathcal{A}} \ell_n^{Logit}(\beta^*)\|_{\max} + \|\nabla_{\mathcal{A}^c} \ell_n^{Logit}(\beta^*)\|_{\max} \\ & < a_1 \lambda.\end{aligned}$$

The desired probability bound can be obtained by using Proposition 4(a) of Fan and Lv (2011) and the union bound. This completes the proof of Theorem 4.  $\square$

## 5.5 Proof of Theorem 5

*Proof.* By definition, it obviously holds that

$$\ell_n^{Logit}(\widehat{\boldsymbol{\beta}}_{Logit}^{lasso}) + \lambda_{lasso} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso}\|_{\ell_1} \leq \ell_n^{Logit}(\boldsymbol{\beta}^*) + \lambda_{lasso} \|\boldsymbol{\beta}^*\|_{\ell_1}.$$

Using the convexity of  $\ell_n^{Logit}(\cdot)$ , we obtain

$$(\nabla \ell_n^{Logit}(\boldsymbol{\beta}^*))'(\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*) + \lambda_{lasso} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso}\|_{\ell_1} \leq \lambda_{lasso} \|\boldsymbol{\beta}^*\|_{\ell_1}.$$

This entails that on the event

$$\left\{ \left\| \frac{1}{n} \mathbf{X}'(\mathbf{y} - \boldsymbol{\mu}(\boldsymbol{\beta}^*)) \right\|_{\max} \leq \frac{1}{2} \lambda_{lasso} \right\} \quad (22)$$

we have

$$-\frac{1}{2} \lambda_{lasso} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*\|_{\ell_1} + \lambda_{lasso} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso}\|_{\ell_1} \leq \lambda_{lasso} \|\boldsymbol{\beta}^*\|_{\ell_1},$$

or

$$\frac{1}{2} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*\|_{\ell_1} \leq \|\boldsymbol{\beta}^*\|_{\ell_1} - \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso}\|_{\ell_1} + \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*\|_{\ell_1}.$$

Using the fact that  $|\beta_j^*| - |\hat{\beta}_j^{lasso}| + |\beta_j^* - \hat{\beta}_j^{lasso}| = 0$  for any  $j \in \mathcal{A}^c$ , we conclude that

$$\frac{1}{2} \|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*\|_{\ell_1} \leq 2 \|\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{lasso} - \boldsymbol{\beta}_{\mathcal{A}}^*\|_{\ell_1}$$

where we denote  $\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} = (\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{lasso}, \widehat{\boldsymbol{\beta}}_{\mathcal{A}^c}^{lasso})$ . The last inequality is equivalent to

$$\|\widehat{\boldsymbol{\beta}}_{\mathcal{A}^c}^{lasso}\|_{\ell_1} \leq 3 \|\widehat{\boldsymbol{\beta}}_{\mathcal{A}}^{lasso} - \boldsymbol{\beta}_{\mathcal{A}}^*\|_{\ell_1}. \quad (23)$$

In what follows, our aim is to derive the upper bound

$$\|\widehat{\boldsymbol{\beta}}_{Logit}^{lasso} - \boldsymbol{\beta}^*\|_{\ell_2} \leq 5\kappa_{Logit}^{-1} s^{1/2} \lambda_{lasso}$$

under the event (22). Then the desired probability bound can be obtained by using the Hoeffding's bound as in the proof of Theorem 4.

Now we consider a map  $F : \mathbb{R}^p \rightarrow \mathbb{R}$  satisfying

$$F(\boldsymbol{\Delta}) = \ell_n^{Logit}(\boldsymbol{\beta}^* + \boldsymbol{\Delta}) - \ell_n^{Logit}(\boldsymbol{\beta}^*) + \lambda_{lasso} (\|\boldsymbol{\beta}^* + \boldsymbol{\Delta}\|_{\ell_1} - \|\boldsymbol{\beta}^*\|_{\ell_1}).$$

In addition, we define  $\widehat{\Delta} = \arg \min_{\Delta} F(\Delta)$ . Then by definition we have  $\widehat{\Delta} = \widehat{\beta}_{Logit}^{lasso} - \beta^*$ . Since  $F(\mathbf{0}) = 0$ ,  $F(\widehat{\Delta}) \leq F(\mathbf{0}) = 0$ . By Lemma 4 of Negahban et al. (2012), because  $\|\widehat{\Delta}_{\mathcal{A}^c}\|_{\ell_1} \leq 3\|\widehat{\Delta}_{\mathcal{A}}\|_{\ell_1}$  as in (23) and convexity of  $F(\Delta)$ , it suffices to show that

$$F(\Delta) > 0$$

for any  $\Delta \in \mathcal{D}$ , where

$$\mathcal{D} = \{\Delta \in \mathbb{R}^p : \|\Delta_{\mathcal{A}^c}\|_{\ell_1} \leq 3\|\Delta_{\mathcal{A}}\|_{\ell_1} \text{ and } \|\Delta\|_{\ell_2} = 5\kappa_{Logit}^{-1}s^{1/2}\lambda_{lasso}\}.$$

To this end, we first obtain a lower bound for  $\|\beta^* + \Delta\|_{\ell_1} - \|\beta^*\|_{\ell_1}$ , i.e.

$$\begin{aligned} \|\beta^* + \Delta\|_{\ell_1} - \|\beta^*\|_{\ell_1} &= \|\beta_{\mathcal{A}}^* + \Delta_{\mathcal{A}}\|_{\ell_1} + \|\Delta_{\mathcal{A}^c}\|_{\ell_1} - \|\beta_{\mathcal{A}}^*\|_{\ell_1} \\ &\geq \|\Delta_{\mathcal{A}^c}\|_{\ell_1} - \|\Delta_{\mathcal{A}}\|_{\ell_1} \end{aligned} \quad (24)$$

Next, we derive a lower bound for  $\ell_n^{Logit}(\beta^* + \Delta) - \ell_n^{Logit}(\beta^*)$ . We define  $G(u) = \ell_n^{Logit}(\beta^* + u\Delta)$ . Recall that  $\psi''(t) = \theta(t)(1 - \theta(t))$  and  $\psi'''(t) = \theta(t)(1 - \theta(t))(2\theta(t) - 1)$  with  $\theta(t) = (1 + \exp(t))^{-1}$ . Then we have

$$\begin{aligned} G''(u) &= \frac{1}{n} \sum_i \psi''(\mathbf{x}'_i(\beta^* + u\Delta)) \cdot (\mathbf{x}'_i\Delta)^2 \\ G'''(u) &= \frac{1}{n} \sum_i \psi'''(\mathbf{x}'_i(\beta^* + u\Delta)) \cdot (\mathbf{x}'_i\Delta)^3 \end{aligned}$$

By using the simple fact that

$$0 \leq |\psi'''(t)| \leq \psi''(t),$$

we have

$$|G'''(u)| \leq \max_i |\mathbf{x}'_i\Delta| \cdot G''(u) \leq m\|\Delta\|_{\ell_1} \cdot G''(u).$$

Note that by the definition of  $\mathcal{D}$ ,

$$\|\Delta\|_{\ell_1} = \|\Delta_{\mathcal{A}}\|_{\ell_1} + \|\Delta_{\mathcal{A}^c}\|_{\ell_1} \leq 4\|\Delta_{\mathcal{A}}\|_{\ell_1} \leq 4ms^{1/2}\|\Delta\|_{\ell_2}.$$

Let  $z = 4ms^{1/2}\|\Delta\|_{\ell_2} = 20m\kappa_{Logit}^{-1}s\lambda_{lasso} > 0$ . Then we have

$$|G'''(u)| \leq zG''(u)$$

By Lemma 1 of Bach (2010), for any convex three times differentiable function  $g(u)$  satisfying  $|g'''(u)| \leq Sg''(u)$  for some  $S > 0$ , we have

$$g(u) - g(0) - g'(0)u \geq g''(0) \cdot S^{-2}\{\exp(-uS) + uS - 1\}.$$

Here we consider  $g(u) = G(u)$  and  $S = z$ . Let  $u = 1$ , and then we obtain

$$G(1) - G(0) - G'(0) \geq G''(0) \cdot h(z), \quad (25)$$

where  $h(z) = z^{-2}(\exp(-z) + z - 1)$ . By simple calculation it can be shown that  $h(z)$  is a decreasing function in  $z > 0$ . Given that  $z \leq 1$  holds by assumption on  $\lambda_{lasso}$ , we have

$$h(z) \geq h(1) = \exp(-1) > 1/3.$$

By definition  $G(1) = \ell_n^{Logit}(\beta^* + \Delta)$ ,  $G(0) = \ell_n^{Logit}(\beta^*)$ ,  $G'(0) = (\nabla \ell_n^{Logit}(\beta^*))' \Delta$  and  $G''(0) = \Delta \nabla^2 \ell_n^{Logit}(\beta^*) \Delta$ . Thus, we can re-write (25) as

$$\begin{aligned} \ell_n^{Logit}(\beta^* + \Delta) - \ell_n^{Logit}(\beta^*) &\geq (\nabla \ell_n^{Logit}(\beta^*))' \Delta + h(z) \Delta \nabla^2 \ell_n^{Logit}(\beta^*) \Delta \\ &> (\nabla \ell_n^{Logit}(\beta^*))' \Delta + \frac{1}{3} \Delta \nabla^2 \ell_n^{Logit}(\beta^*) \Delta \end{aligned} \quad (26)$$

Next, under the event  $\{\|\frac{1}{n} \mathbf{X}'(\mathbf{y} - \mu(\beta^*))\|_{\max} \leq \frac{1}{2} \lambda_{lasso}\}$ , we have

$$(\nabla \ell_n^{Logit}(\beta^*))' \Delta \geq -\frac{1}{2} \lambda_{lasso} \|\Delta\|_{\ell_1}. \quad (27)$$

Now under the same event, we combine (24), (26), (27) and the restricted eigenvalue condition (C2) to obtain

$$\begin{aligned} F(\Delta) &> \frac{1}{3} \kappa_{Logit} \|\Delta\|_{\ell_2}^2 - \frac{1}{2} \lambda_{lasso} \|\Delta\|_{\ell_1} + \lambda_{lasso} (\|\Delta_{\mathcal{A}^c}\|_{\ell_1} - \|\Delta_{\mathcal{A}}\|_{\ell_1}) \\ &\geq \frac{1}{3} \kappa_{Logit} \|\Delta\|_{\ell_2}^2 - \frac{3}{2} \lambda_{lasso} \|\Delta_{\mathcal{A}}\|_{\ell_1} \\ &\geq \frac{1}{3} \kappa_{Logit} \|\Delta\|_{\ell_2}^2 - \frac{3}{2} \lambda_{lasso} \cdot s^{1/2} \|\Delta\|_{\ell_2} \\ &= \frac{5s\lambda_{lasso}^2}{6\kappa_{Logit}} \\ &> 0. \end{aligned}$$

This completes the proof of Theorem 5. □

## 5.6 Proof of Theorem 6

*Proof.* We first derive an upper bound for  $\delta_2 = \Pr(\|\hat{\Theta}_G^{oracle}\|_{\min} \leq a\lambda)$ . A translation of (3) into the precision matrix estimation setting becomes

$$\hat{\Sigma}_{\mathcal{A}}^{oracle} = \hat{\Sigma}_{\mathcal{A}}^n.$$

Let  $\Sigma^\Delta = (\Theta^\star + \Delta)^{-1}$ . Define a map  $F : \mathbb{B}(r) \subset \mathbb{R}^{p^2} \longrightarrow \mathbb{R}^{p^2}$  such that

$$F(\text{vec}(\Delta)) = ((F_{\mathcal{A}}(\text{vec}(\Delta_{\mathcal{A}})))', \mathbf{0}')'$$

with

$$F_{\mathcal{A}}(\text{vec}(\Delta_{\mathcal{A}})) = (\mathbf{H}_{\mathcal{A}\mathcal{A}}^\star)^{-1} \cdot (\text{vec}(\Sigma_{\mathcal{A}}^\Delta) - \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^n)) + \text{vec}(\Delta_{\mathcal{A}}) \quad (28)$$

and the convex compact set

$$\mathbb{B}(r) = \{\Delta : \|\Delta_{\mathcal{A}}\|_{\max} \leq r, \Delta_{\mathcal{A}^c} = \mathbf{0}\},$$

where  $r = 2K_2 \cdot \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^\star\|_{\max}$ . We will show that

$$F(\mathbb{B}(r)) \subset \mathbb{B}(r) \quad (29)$$

under the condition

$$\|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^\star\|_{\max} < \min\left\{\frac{1}{6K_1K_2d}, \frac{1}{6K_1^3K_2^2d}\right\}. \quad (30)$$

If (29) holds, an application of the Brouwer's fixed point theorem yields that there exists a fixed point  $\widehat{\Delta} \in \mathbb{B}(r)$  satisfying

$$F_{\mathcal{A}}(\text{vec}(\widehat{\Delta}_{\mathcal{A}})) = \text{vec}(\widehat{\Delta}_{\mathcal{A}}) \quad \text{and} \quad \widehat{\Delta}_{\mathcal{A}^c} = \mathbf{0}.$$

In other words,  $\widehat{\Delta}_{\mathcal{A}} = \widehat{\Theta}_{\mathcal{A}}^{\text{oracle}} - \Theta_{\mathcal{A}}^\star$  by the uniqueness and thus

$$\|\widehat{\Theta}^{\text{oracle}} - \Theta^\star\|_{\max} = \|\widehat{\Delta}\|_{\max} \leq r. \quad (31)$$

We now establish (29). For any  $\Delta \in \mathbb{B}(r)$ , we have

$$\|\Sigma^\star \Delta\|_{\ell_\infty} \leq K_1 \cdot \|\Delta\|_{\ell_1} \leq K_1 \cdot dr = 2K_1K_2d \cdot \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^\star\|_{\max} < \frac{1}{3},$$

by using (30). Thus,

$$\mathbf{J} = \sum_{j=0}^{\infty} (-1)^j (\Sigma^\star \Delta)^j$$

is a convergent matrix series of  $\Delta$ . Hence,

$$\Sigma^\Delta = (\mathbf{I} + \Sigma^\star \Delta)^{-1} \cdot \Sigma^\star = \Sigma^\star - \Sigma^\star \Delta \Sigma^\star + \mathbf{R}^\Delta, \quad (32)$$

where  $\mathbf{R}^\Delta = (\Sigma^\star \Delta)^2 \cdot \mathbf{J} \Sigma^\star$ . Then it immediately yields that

$$\text{vec}(\Sigma_{\mathcal{A}}^\Delta) - \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^n) = (\text{vec}(\Sigma_{\mathcal{A}}^\star) - \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^n)) - \text{vec}(\Sigma_{\mathcal{A}}^\star \Delta \Sigma_{\mathcal{A}}^\star) + \text{vec}(\mathbf{R}_{\mathcal{A}}^\Delta). \quad (33)$$

Note that

$$\Sigma^* \Delta \Sigma^* = (\Sigma^* \otimes \Sigma^*) \cdot \text{vec}(\Delta) = \mathbf{H}^* \cdot \text{vec}(\Delta)$$

and hence

$$\text{vec}(\Sigma^* \Delta \Sigma^*) = \mathbf{H}_{\mathcal{A}\mathcal{A}}^* \cdot \text{vec}(\Delta_{\mathcal{A}}).$$

Now we follow the same lines of the proof as in Lemma 5 of Ravikumar et al. (2008) to obtain

$$\|\mathbf{R}^\Delta\|_{\max} = \max_{(i,j)} |\mathbf{e}'_i ((\Sigma^* \Delta)^2 \cdot \mathbf{J} \Sigma^*) \mathbf{e}_j| \leq \frac{3}{2} K_1^3 \cdot d \|\Delta\|_{\max}^2. \quad (34)$$

Therefore, a combination of (28), (33) and (34) yields the following upper bound,

$$\begin{aligned} & \|F_{\mathcal{A}}(\text{vec}(\Delta_{\mathcal{A}}))\|_{\max} \\ &= \|(\mathbf{H}_{\mathcal{A}\mathcal{A}}^*)^{-1} \cdot (\text{vec}(\Sigma_{\mathcal{A}}^*) - \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^n)) + \text{vec}(\mathbf{R}_{\mathcal{A}}^\Delta)\|_{\max} \\ &\leq K_2 \cdot (\|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} + \|\mathbf{R}_{\mathcal{A}}^\Delta\|_{\max}) \\ &\leq r. \end{aligned}$$

This proves (29).

Under the additional condition

$$\|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} < \frac{1}{2K_2} (\|\Theta_{\mathcal{A}}^*\|_{\min} - a\lambda),$$

by (31) and the definition of  $r$ , we have that

$$\begin{aligned} \|\widehat{\Theta}_{\mathcal{A}}^{oracle}\|_{\min} &\geq \|\Theta_{\mathcal{A}}^*\|_{\min} - \|\widehat{\Theta}^{oracle} - \Theta^*\|_{\max} \\ &= \|\Theta_{\mathcal{A}}^*\|_{\min} - 2K_2 \cdot \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} \\ &> a\lambda. \end{aligned}$$

Thus,

$$\delta_2 \leq \Pr \left( \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} > \frac{1}{2K_2} \min \left\{ \frac{1}{3K_1 d}, \frac{1}{3K_1^3 K_2 d}, \|\Theta_{\mathcal{A}}^*\|_{\min} - a\lambda \right\} \right)$$

An application of (8) yields the bound on  $\delta_2$ .

We now deal with  $\delta_1 = \Pr(\|\nabla_{\mathcal{A}^c} \ell_n(\widehat{\Theta}_G^{oracle})\|_{\max} \geq a_1 \lambda)$ . Note that

$$\nabla_{\mathcal{A}^c} \ell_n(\widehat{\Theta}_G^{oracle}) = \widehat{\Sigma}_{\mathcal{A}^c}^n - \widehat{\Sigma}_{\mathcal{A}^c}^{oracle}$$

and hence

$$\|\nabla_{\mathcal{A}^c} \ell_n(\widehat{\Theta}_G^{oracle})\|_{\max} \leq \|\widehat{\Sigma}_{\mathcal{A}^c}^n - \Sigma_{\mathcal{A}^c}^*\|_{\max} + \|\widehat{\Sigma}_{\mathcal{A}^c}^{oracle} - \Sigma_{\mathcal{A}^c}^*\|_{\max}. \quad (35)$$

Note  $\|\widehat{\Sigma}_{\mathcal{A}^c}^n - \Sigma_{\mathcal{A}^c}^*\|_{\max}$  is bounded by using (8). Then we only need to bound  $\|\widehat{\Sigma}_{\mathcal{A}^c}^{oracle} - \Sigma_{\mathcal{A}^c}^*\|_{\max}$ . Recall  $\widehat{\Delta} = \widehat{\Theta}^{oracle} - \Theta^*$ . By (32), we have

$$\widehat{\Sigma}^{oracle} = (\Theta^* + \widehat{\Delta})^{-1} = (\mathbf{I} + \Sigma^* \Delta)^{-1} \cdot \Sigma^* = \Sigma^* - \Sigma^* \widehat{\Delta} \Sigma^* + \widehat{\mathbf{R}}. \quad (36)$$

where  $\widehat{\mathbf{R}} = (\Sigma^* \widehat{\Delta})^2 \cdot \widehat{\mathbf{J}} \Sigma^*$  and  $\widehat{\mathbf{J}}$  is defined similarly to  $\mathbf{J}$  with  $\Delta$  replaced by  $\widehat{\Delta}$ . Then  $\widehat{\mathbf{J}}$  is a convergent matrix series under condition (30). In terms of  $\mathcal{A}$ , we can equivalently write (36) as

$$\begin{aligned} \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^{oracle}) - \text{vec}(\Sigma_{\mathcal{A}}^*) &= -\mathbf{H}_{\mathcal{A}\mathcal{A}}^* \cdot \text{vec}(\widehat{\Delta}_{\mathcal{A}}) + \text{vec}(\widehat{\mathbf{R}}_{\mathcal{A}}) \\ \text{vec}(\widehat{\Sigma}_{\mathcal{A}^c}^{oracle}) - \text{vec}(\Sigma_{\mathcal{A}^c}^*) &= -\mathbf{H}_{\mathcal{A}^c\mathcal{A}}^* \cdot \text{vec}(\widehat{\Delta}_{\mathcal{A}}) + \text{vec}(\widehat{\mathbf{R}}_{\mathcal{A}^c}) \end{aligned}$$

where we use the fact that  $\widehat{\Delta}_{\mathcal{A}^c} = \mathbf{0}$ . Solving  $\text{vec}(\widehat{\Delta}_{\mathcal{A}})$  from the first equation and substituting it into the second equation, we obtain

$$\begin{aligned} &\text{vec}(\widehat{\Sigma}_{\mathcal{A}^c}^{oracle}) - \text{vec}(\Sigma_{\mathcal{A}^c}^*) \\ &= \mathbf{H}_{\mathcal{A}^c\mathcal{A}}^* (\mathbf{H}_{\mathcal{A}\mathcal{A}}^*)^{-1} \cdot \left( \text{vec}(\widehat{\Sigma}_{\mathcal{A}}^{oracle}) - \text{vec}(\Sigma_{\mathcal{A}}^*) - \text{vec}(\widehat{\mathbf{R}}_{\mathcal{A}}) \right) + \text{vec}(\widehat{\mathbf{R}}_{\mathcal{A}^c}) \end{aligned}$$

Recall (34) holds under condition (30). Thus, we have

$$\|\widehat{\mathbf{R}}\|_{\max} \leq \frac{3}{2} K_1^3 \cdot d \|\widehat{\Delta}\|_{\max}^2 = 6 K_1^3 K_2^2 d \cdot \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} \leq \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max}.$$

Thus under the additional event

$$\left\{ \|\widehat{\Sigma}_{\mathcal{A}^c}^n - \Sigma_{\mathcal{A}^c}^*\|_{\max} < \frac{a_1 \lambda}{2} \right\} \cap \left\{ \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} \leq \frac{a_1 \lambda}{4K_3 + 2} \right\},$$

we derive the desired upper bound for (35) by using the triangular inequality,

$$\begin{aligned} \|\nabla_{\mathcal{A}^c} \ell_n(\widehat{\Theta}_G^{oracle})\|_{\max} &\leq \|\widehat{\Sigma}_{\mathcal{A}^c}^n - \Sigma_{\mathcal{A}^c}^*\|_{\max} + \|\Sigma_{\mathcal{A}^c}^* - \widehat{\Sigma}_{\mathcal{A}^c}^{oracle}\|_{\max} \\ &\leq \frac{a_1 \lambda}{2} + (2K_3 + 1) \cdot \|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} \\ &< a_1 \lambda. \end{aligned}$$

Therefore,

$$\begin{aligned} \delta_1 &\leq \Pr\{\|\widehat{\Sigma}_{\mathcal{A}}^n - \Sigma_{\mathcal{A}}^*\|_{\max} \geq \min\{\frac{1}{6K_1 K_2 d}, \frac{1}{6K_1^3 K_2^2 d}, \frac{a_1 \lambda}{4K_3 + 2}\}\} \\ &\quad + \Pr\{\|\widehat{\Sigma}_{\mathcal{A}^c}^n - \Sigma_{\mathcal{A}^c}^*\|_{\max} > \frac{a_1 \lambda}{2}\}. \end{aligned}$$

An application of (8) yields the bound on  $\delta_1$ . This completes the proof of Theorem 6.  $\square$



## References

- Bach, F. (2010), Self-concordant analysis for logistic regression, *Electronic Journal of Statistics*, 4, 384–414.
- Bickel, P. and Levina, E. (2008), Regularized estimation of large covariance matrices, *The Annals of Statistics*, 36, 199–227.
- Bickel, P., Ritov, Y. and Tsybakov, A. (2009), Simultaneous analysis of Lasso and Dantzig selector, *The Annals of Statistics*, 37, 1705–1732.
- Bradic, J., Fan, J. and Jiang, J. (2012), Regularization for Cox’s proportional hazards model with NP-dimensionality, *The Annals of Statistics*, 39, 3092–3120.
- Cai, T., Liu, W. and Luo, X. (2011), A constrained  $\ell_1$  minimization approach to sparse precision matrix estimation, *Journal of the American Statistical Association*, 106, 594–607.
- Cai, T., Liu, W. and Luo, X. (2012), clime: an R package for the constrained  $\ell_1$ -minimization for inverse covariance matrix estimation, Available from <http://cran.r-project.org/web/packages/clime/index.html>.
- Candes, E., Wakin, M. and Boyd, S. (2008), Enhancing sparsity by reweighted  $\ell_1$  minimization, *Journal of Fourier Analysis and Applications*, 14, 877–905.
- Candes, E. and Tao, T. (2007), The Dantzig selector: Statistical estimation when  $p$  is much larger than  $n$ , *The Annals of statistics*, 35(6), 2313–2351.
- Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004), Least angle regression (with discussion), *The Annals of statistics*, 32, 407–499.
- Fan, J. and Li, R. (2001), Variable selection via nonconcave penalized likelihood and its oracle properties, *Journal of the American Statistical Association*, 96, 1348–1360.
- Fan, J. and Lv, J. (2011), Non-concave penalized likelihood with NP-dimensionality, *IEEE Transactions on Information Theory*, 57, 5467–5484.

- Fan, J. and Peng, H. (2004), Nonconcave penalized likelihood with a diverging number of parameters, *The Annals of Statistics*, 32, 928–961.
- Fan, J. and Song, R. (2010). Sure independence screening in generalized linear models with NP-dimensionality. *The Annals of Statistics*, 38, 3567–3604.
- Friedman, J., Hastie, T. and Tibshirani, R. (2008), Sparse inverse covariance estimation with the graphical lasso, *Biostatistics*, 9, 432–441.
- Friedman, J., Hastie, T. and Tibshirani, R. (2010), Regularization paths for generalized linear models via coordinate descent, *Journal of Statistical Software*, 33, 1–22.
- Friedman, J., Hastie, T. and Tibshirani R. (2011), glasso – an R package for the estimation of Gaussian graphical models via the Graphical Lasso, *Available from* <http://cran.r-project.org/web/packages/glasso/index.html>.
- Friedman, J., Hastie, T. and Tibshirani R. (2012), glmnet – an R package for the Lasso and elastic-net regularized generalized linear models, *Available from* <http://cran.r-project.org/web/packages/glmnet/index.html>.
- Huang, J. and Zhang, C. (2012), Estimation and selection via absolute penalized convex minimization and its multistage adaptive applications, *Journal of Machine Learning Research*, 12, 1839–1864.
- Hunter, D. and Lange, K. (2004), A tutorial on MM algorithms, *The American Statistician*, 58, 30–37.
- Hunter, D. and Li, R. (2005), Variable selection using MM algorithms, *The Annals of Statistics*, 33, 1617–1642.
- Kim Y, Choi H, Oh, H.S. (2008). Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association*, 103, 1665–1673.
- Lam, C. and Fan, J. (2009), Sparsistency and rates of convergence in large covariance matrix estimation, *The Annals of Statistics*, 37, 4254–4278.
- Mazumder, R., Friedman, J. and Hastie, T. (2011), SparseNet: Coordinate descent with non-convex penalties, *Journal of the American Statistical Association*, 106, 1125–1138.

- Meinshausen, N. and Bühlmann, P. (2006), High-dimensional graphs and variable selection with the lasso, *The Annals of Statistics*, 34, 1436–1462.
- Negahban, S., Ravikumar, P., Wainwright, M. and Yu, B. (2012), A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers, *Statistical Science*, to appear.
- Raskutti, G., Wainwright, M. and Yu, B. (2010), Restricted eigenvalue properties for correlated gaussian designs, *Journal of Machine Learning Research*, 11, 2241–2259.
- Ravikumar, P., Wainwright, M. and Lafferty, J. (2010), High-dimensional Ising model selection using  $\ell_1$ -regularized logistic regression, *The Annals of Statistics*, 38, 1287–1319.
- Ravikumar, P., Wainwright, M., Raskutti, G. and Yu, B. (2008), High-dimensional covariance estimation by minimizing  $\ell_1$ -penalized log-determinant divergence, *Advances in Neural Information Processing Systems*.
- Saulis, L. and Statulevicius, V. (1991), *Limit theorems for large deviations*, Springer.
- Tibshirani, R. (1996), Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society, Series B*, 58, 267–288.
- Van De Geer, S. (2008), High-dimensional generalized linear models and the Lasso, *The Annals of Statistics*, 36, 614–645.
- Van De Geer, S. and Bühlmann, P. (2009), On the conditions used to prove oracle results for the Lasso, *Electronic Journal of Statistics*, 3, 1360–1392.
- Wainwright, M. (2009), Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (Lasso), *IEEE Transactions on Information Theory*, 55, 2183–2202.
- Xue, L., Zou, H. and Cai, T. (2012), Non-concave penalized composite conditional likelihood estimation of sparse Ising models, *The Annals of Statistics*, 40, 1403–1429.
- Yuan, M. and Lin, Y. (2007), Model selection and estimation in the Gaussian graphical model, *Biometrika*, 94, 19–35.

- Zhang, C. (2010a), Nearly unbiased variable selection under minimax concave penalty, *The Annals of Statistics*, 38, 894–942.
- Zhang, C. and Zhang, T. (2012), A general theory of concave regularization for high dimensional sparse estimation problems, *Statistical Science*, to appear.
- Zhang, T. (2010b), Analysis of multi-stage convex relaxation for sparse regularization, *Journal of Machine Learning Research*, 11, 1081–1107.
- Zhao, P. and Yu, B. (2006), On model selection consistency of Lasso, *Journal of Machine Learning Research*, 7, 2541–2563.
- Zou, H. (2006), The adaptive lasso and its oracle properties, *Journal of the American Statistical Association*, 101, 1418–1429.
- Zou, H. and Li, R. (2008), One-step sparse estimates in nonconcave penalized likelihood models (with discussion), *The Annals of Statistics*, 36, 1509–1566.

Table 1: Numerical comparison of LASSO, SCAD & MCP for the sparse linear regression problem in Model 1. Estimation performance is measured by the  $\ell_1$  loss, and selection accuracy is measured by counts of false negative ( $\#FN$ ) or false positive ( $\#FP$ ). Each metric is averaged over 100 replications with its standard error shown in the parenthesis.

Method	$n = 100 \ \& \ p = 500$			$n = 100 \ \& \ p = 1000$		
	$\ell_1$ loss	$\#FP$	$\#FN$	$\ell_1$ loss	$\#FP$	$\#FN$
LASSO	1.040	11.36	0	1.204	14.68	0
	(0.038)	(0.63)	(0)	(0.045)	(0.74)	(0)
SCAD-cd	0.333	1.69	0	0.339	2.22	0
	(0.018)	(0.33)	(0)	(0.017)	(0.40)	(0)
SCAD-lla0	0.268	0	0	0.293	0	0
	(0.012)	(0)	(0)	(0.014)	(0)	(0)
SCAD-lla*	0.267	0	0	0.291	0	0
	(0.012)	(0)	(0)	(0.014)	(0)	(0)
MCP-cd	0.333	0.77	0	0.314	0.75	0
	(0.018)	(0.16)	(0)	(0.015)	(0.14)	(0)
MPC-lla0	0.290	0	0	0.295	0	0
	(0.015)	(0)	(0)	(0.016)	(0)	(0)
MCP-lla*	0.288	0	0	0.290	0	0
	(0.014)	(0)	(0)	(0.015)	(0)	(0)

Table 2: Numerical comparison of LASSO, SCAD & MCP for the sparse linear regression problem in Model 2. Estimation performance is measured by the  $\ell_1$  loss, and selection accuracy is measured by counts of false negative ( $\#FN$ ) or false positive ( $\#FP$ ). Each metric is averaged over 100 replications with its standard error shown in the parenthesis.

Method	$n = 100 \ \& \ p = 500$			$n = 100 \ \& \ p = 1000$		
	$\ell_1$ loss	$\# \text{ FP}$	$\# \text{ FN}$	$\ell_1$ loss	$\# \text{ FP}$	$\# \text{ FN}$
LASSO	4.844	40.28	0	6.829	53.27	0.03
	(0.135)	(1.06)	(0)	(0.171)	(1.23)	(0.01)
SCAD-cd	1.227	8.78	0	1.288	11.25	0
	(0.036)	(0.59)	(0)	(0.042)	(0.64)	(0)
SCAD-lla0	0.914	0	0.04	1.093	0.15	0.15
	(0.033)	(0)	(0.02)	(0.074)	(0.06)	(0.04)
SCAD-lla*	0.903	0	0.03	1.064	0.10	0.15
	(0.031)	(0)	(0.01)	(0.063)	(0.04)	(0.04)
MCP-cd	0.948	1.16	0	1.149	1.29	0.14
	(0.028)	(0.17)	(0)	(0.131)	(0.18)	(0.08)
MPC-lla0	0.941	0.17	0.01	1.052	0.28	0.07
	(0.033)	(0.06)	(0.01)	(0.067)	(0.10)	(0.03)
MCP-lla*	0.928	0.13	0.01	1.031	0.23	0.07
	(0.033)	(0.05)	(0.01)	(0.064)	(0.09)	(0.03)

Table 3: Numerical comparison of LASSO, SCAD & MCP for the sparse logistic regression problem in Model 3. Estimation performance is measured by the  $\ell_1$  loss, and selection accuracy is measured by counts of false negative ( $\#FN$ ) or false positive ( $\#FP$ ). Each metric is averaged over 100 replications with its standard error shown in the parenthesis.

Method	$n = 200 \text{ \& } p = 500$			$n = 200 \text{ \& } p = 1000$		
	$\ell_1$ loss	$\#FP$	$\#FN$	$\ell_1$ loss	$\#FP$	$\#FN$
LASSO	5.274	20.30	0.01	5.670	24.02	0.04
	(0.047)	(0.39)	(0.01)	(0.049)	(0.44)	(0.01)
SCAD-cd	4.086	10.79	0.04	4.496	13.99	0.08
	(0.054)	(0.25)	(0.01)	(0.056)	(0.31)	(0.01)
SCAD-lla0	1.851	0.31	0.09	2.159	0.31	0.22
	(0.092)	(0.04)	(0.02)	(0.108)	(0.05)	(0.02)
SCAD-lla*	1.822	0.24	0.10	2.080	0.26	0.19
	(0.090)	(0.04)	(0.02)	(0.103)	(0.04)	(0.02)
MCP-cd	2.671	2.23	0.27	2.936	2.64	0.47
	(0.056)	(0.09)	(0.03)	(0.074)	(0.11)	(0.03)
MPC-lla0	1.880	0.30	0.12	2.159	0.45	0.19
	(0.093)	(0.04)	(0.02)	(0.108)	(0.06)	(0.02)
MCP-lla*	1.848	0.26	0.09	2.146	0.35	0.23
	(0.089)	(0.04)	(0.02)	(0.097)	(0.05)	(0.03)

Table 4: Numerical comparison of LASSO, SCAD & MCP for the sparse logistic regression problem in Model 4. Estimation performance is measured by the  $\ell_1$  loss, and selection accuracy is measured by counts of false negative ( $\#FN$ ) or false positive ( $\#FP$ ). Each metric is averaged over 100 replications with its standard error shown in the parenthesis.

Method	$n = 200 \text{ \& } p = 500$			$n = 200 \text{ \& } p = 1000$		
	$\ell_1$ loss	$\#FP$	$\#FN$	$\ell_1$ loss	$\#FP$	$\#FN$
LASSO	13.909	49.56	0.22	15.079	55.92	0.59
	(0.053)	(0.62)	(0.03)	(0.061)	(0.93)	(0.04)
SCAD-cd	7.906	20.30	0.42	9.123	27.72	0.58
	(0.129)	(0.41)	(0.04)	(0.147)	(0.46)	(0.04)
SCAD-lla0	5.612	0.90	1.50	6.416	0.79	2.73
	(0.159)	(0.08)	(0.07)	(0.129)	(0.06)	(0.08)
SCAD-lla*	5.209	0.44	1.54	6.413	0.74	2.74
	(0.128)	(0.05)	(0.07)	(0.143)	(0.06)	(0.09)
MCP-cd	6.227	3.10	1.38	6.973	3.62	1.46
	(0.121)	(0.14)	(0.04)	(0.160)	(0.14)	(0.08)
MPC-lla0	6.168	1.18	1.44	6.884	1.11	2.81
	(0.168)	(0.08)	(0.07)	(0.141)	(0.09)	(0.09)
MCP-lla*	5.854	0.86	1.46	6.300	0.78	2.64
	(0.267)	(0.07)	(0.07)	(0.135)	(0.07)	(0.08)



Table 5: Numerical comparison of GLASSO, CLIME, GSCAD & GMCP for the sparse precision matrix estimation problem in Model 5. Estimation performance is measured by the Operator norm and the Frobenius norm, and selection accuracy is measured by counts of false negative (#FN) or false positive (#FP).

Method	Operator norm	Frobenius norm	# FP	# FN
$n = 100 \text{ \& } p = 100$				
GLASSO	1.452	6.115	743.56	1.34
	(0.009)	(0.022)	(10.75)	(0.17)
CLIME	1.401	5.885	741.16	2.42
	(0.012)	(0.029)	(12.80)	(0.24)
GSCAD-lla0	1.163	4.420	641.82	1.96
	(0.019)	(0.029)	(9.41)	(0.20)
GSCAD-lla*	1.162	4.416	635.49	1.94
	(0.019)	(0.029)	(9.39)	(0.19)
GMCP-lla0	1.527	4.556	291.04	6.45
	(0.038)	(0.042)	(5.12)	(0.32)
GMCP-lla*	1.391	4.310	229.87	6.29
	(0.031)	(0.037)	(4.56)	(0.33)
$n = 200 \text{ \& } p = 100$				
GLASSO	1.270	5.424	366.44	0.04
	(0.005)	(0.013)	(3.44)	(0.02)
CLIME	0.962	3.923	390.34	0.06
	(0.007)	(0.016)	(4.35)	(0.02)
GSCAD-lla0	0.772	2.793	285.15	0.26
	(0.010)	(0.013)	(3.05)	(0.02)
GSCAD-lla*	0.746	2.514	285.13	0.06
	(0.007)	(0.012)	(3.05)	(0.02)
GMCP-lla0	0.755	2.517	180.85	0.32
	(0.009)	(0.015)	(3.17)	(0.05)
GMCP-lla*	0.725	2.468	152.06	0.38
	(0.008)	(0.011)	(2.31)	(0.05)

Table 6: Numerical comparison of GLASSO, CLIME, GSCAD & GMCP for the sparse precision matrix estimation problem in Model 6. Estimation performance is measured by the Operator norm and the Frobenius norm, and selection accuracy is measured by counts of false negative (#FN) or false positive (#FP).

Method	Operator norm	Frobenius norm	# FP	# FN
$n = 100 \text{ \& } p = 100$				
GLASSO	11.631	25.447	236.76	56.16
	(0.015)	(0.032)	(5.19)	(0.52)
CLIME	8.558	18.404	323.04	12.26
	(0.053)	(0.075)	(7.22)	(0.38)
GSCAD-lla0	10.727	20.683	228.70	54.54
	(0.048)	(0.121)	(4.92)	(0.58)
GSCAD-lla*	6.416	13.363	196.60	30.02
	(0.126)	(0.153)	(5.27)	(0.57)
GMCP-lla0	10.337	19.202	200.37	52.24
	(0.071)	(0.120)	(4.26)	(0.60)
GMCP-lla*	5.977	12.741	44.79	25.18
	(0.167)	(0.169)	(3.42)	(0.61)
$n = 200 \text{ \& } p = 100$				
GLASSO	11.492	24.857	78.18	49.74
	(0.009)	(0.021)	(1.51)	(0.33)
CLIME	5.539	12.328	350.38	1.92
	(0.056)	(0.061)	(5.21)	(0.12)
GSCAD-lla0	10.411	18.820	76.16	47.62
	(0.034)	(0.092)	(1.47)	(0.34)
GSCAD-lla*	3.739	7.633	67.00	9.58
	(0.059)	(0.074)	(1.65)	(0.28)
GMCP-lla0	9.937	16.813	75.61	45.08
	(0.053)	(0.076)	(1.41)	(0.33)
GMCP-lla*	3.406	7.160	14.64	6.36
	(0.054)	(0.070)	(1.20)	(0.27)